Scalable Algebraic Multigrid on Blue Gene/L

Hans De Sterck, Jeff Butler

Department of Applied Mathematics

University of Waterloo

Ulrike Meier Yang

Center for Applied Scientific Computing

Lawrence Livermore National Lab, USA

University of Waterloo

CAIMS Annual Meeting,
Toronto, 19 June 2006

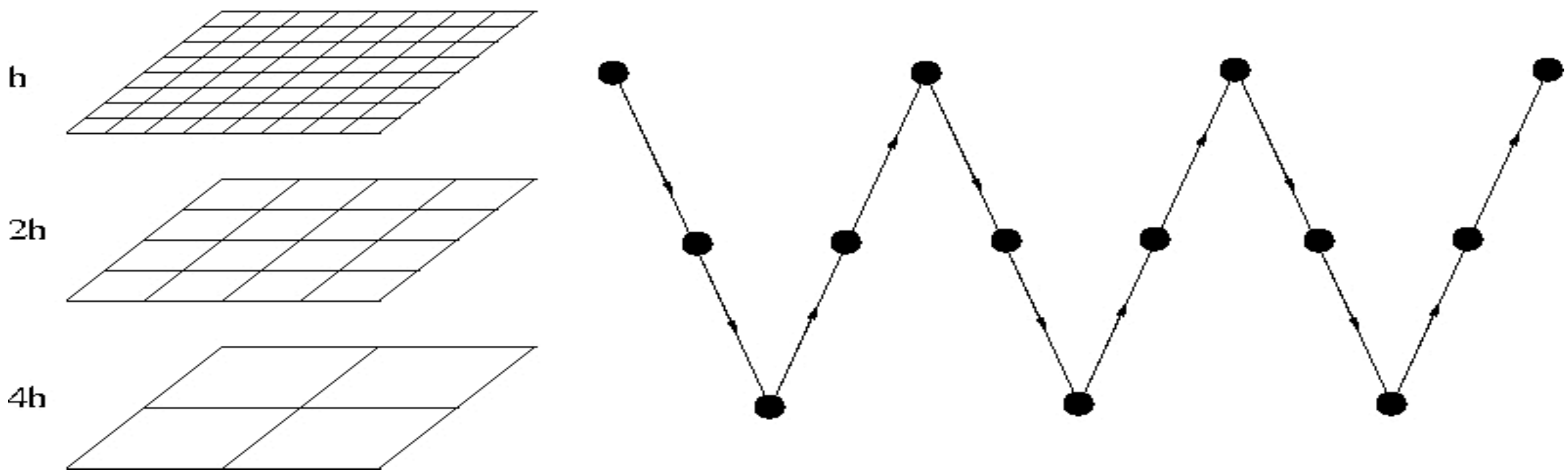
Outline

1. introduction: algebraic multigrid (AMG)
2. classical coarsening may lead to complexity growth
3. Parallel Modified Independent Set (PMIS) coarsening
4. improving interpolation
5. scaling results on Blue Gene/L

(1) introduction: algebraic multigrid (AMG)

- *solve* $\mathbf{Au} = \mathbf{f}$
- $\mathbf{A}$ *from 3D PDE – sparse!*
- *large problems (10^9 dof) - parallel*
- *unstructured grid problems*

algebraic multigrid (AMG)



- *multi-level*
- *iterative*
- *algebraic: suitable for unstructured grids!*

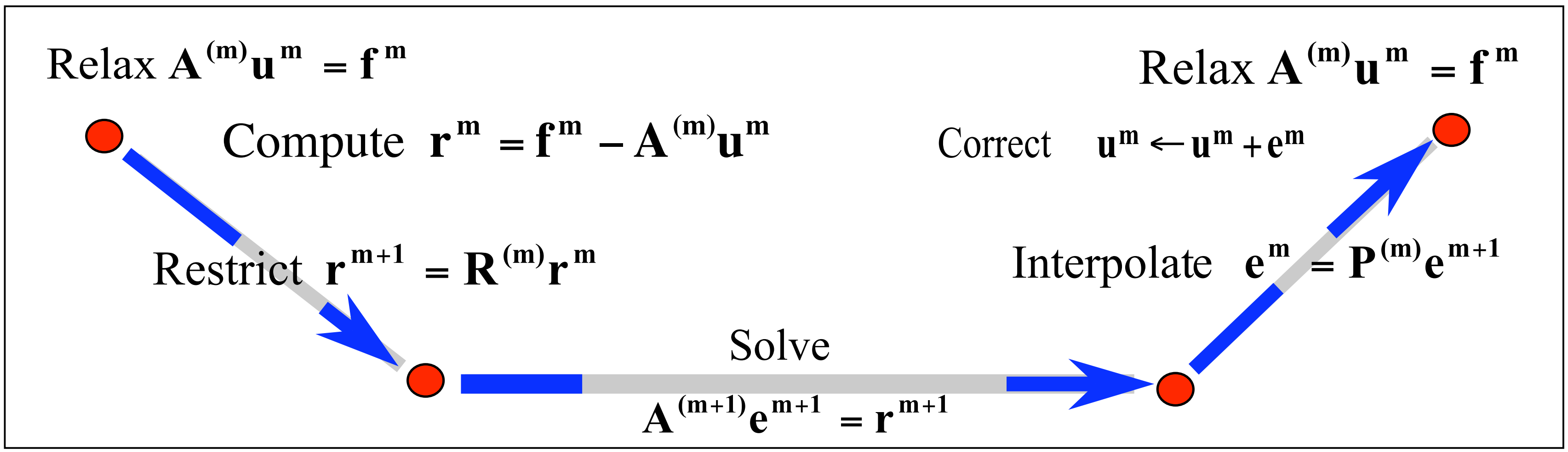
AMG building blocks

Setup Phase:

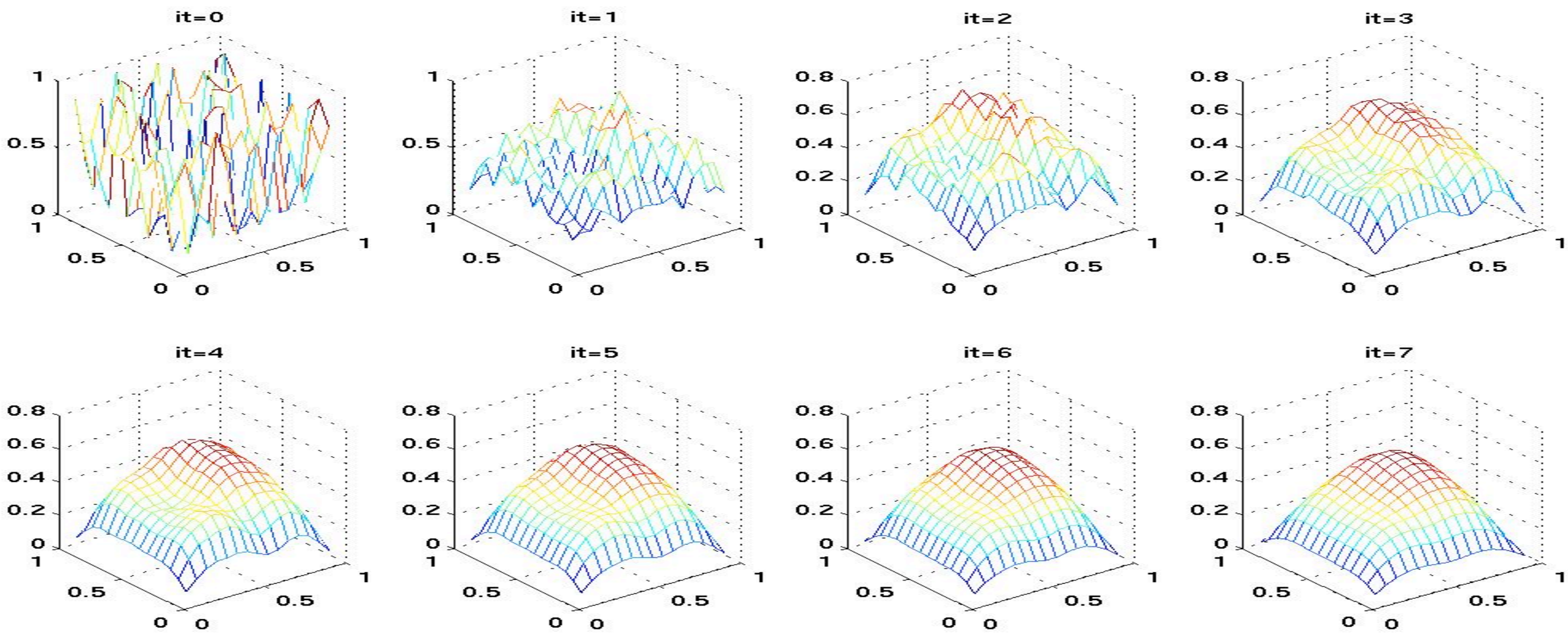
- Select coarse "grids"
- Define interpolation, $P^{(m)}, \quad m = 1,2,\ldots$
- Define restriction and coarse-grid operators

$$R^{(m)} = P^{(m)T} \qquad A^{(m+1)} = P^{(m)T} A^{(m)} P^{(m)}$$

Solve Phase



2D model problem: $-u_{xx} - u_{yy} = f(x, y)$



- *high-frequency error is removed by relaxation*
- *low-frequency error needs to be removed by coarse-grid correction*
- *low-frequency error on fine grid becomes higher frequency error on coarse grid*

AMG complexity - scalability

- *Operator complexity* $C_{op} = \frac{\sum_i \text{nonzeros}(A_i)}{\text{nonzeros}(A_0)}$

e.g., 3D, ideally: $C_{op} = 1 + 1/8 + 1/64 + \ldots < 8/7$

measure of memory use, and work in solve phase

- *scalable algorithm:*

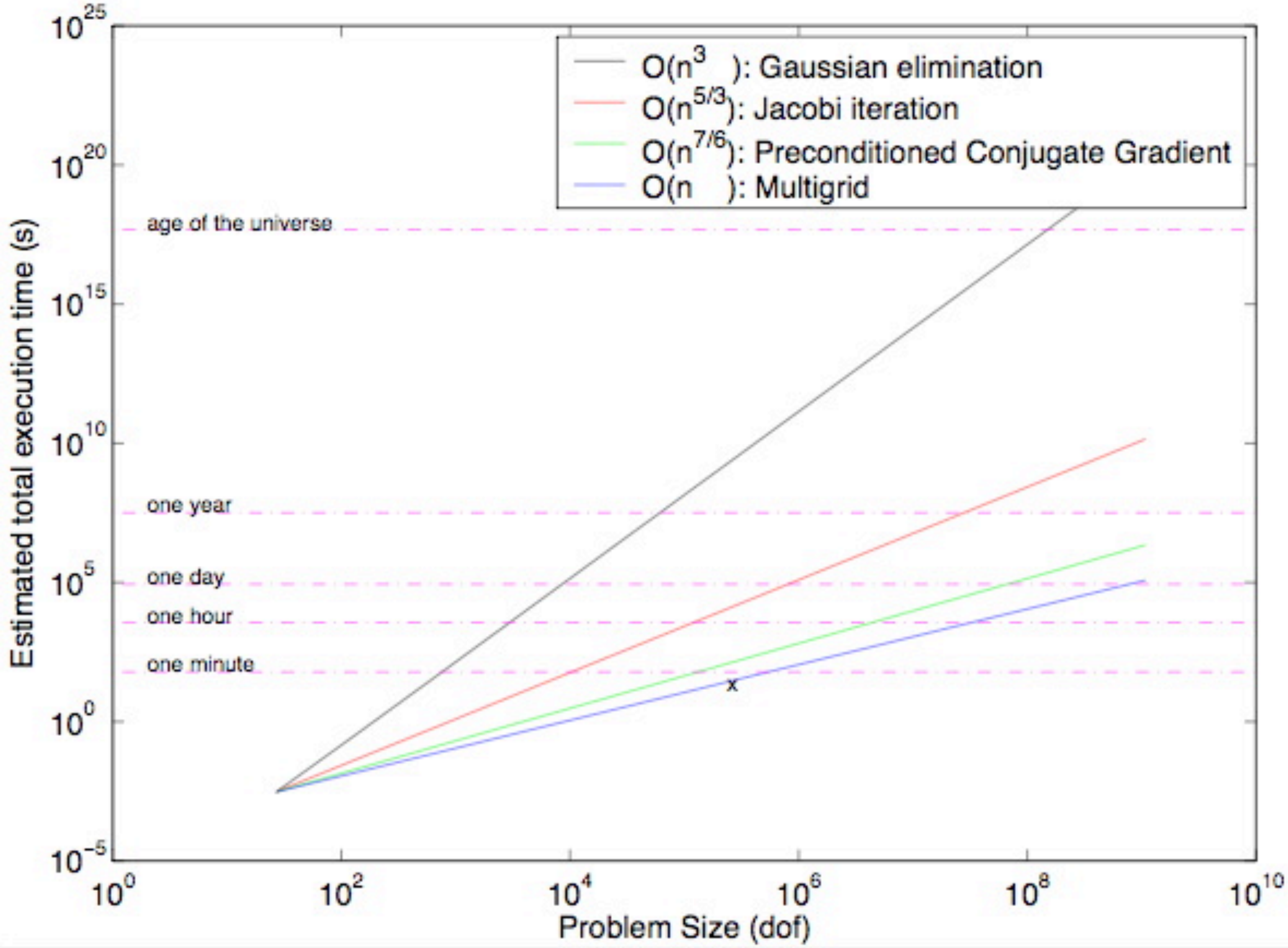
O(n) operations per V-cycle (C_{op} bounded)

AND

number of V-cycles independent of n

(ρ_{AMG} independent of n)

O(n) scalability



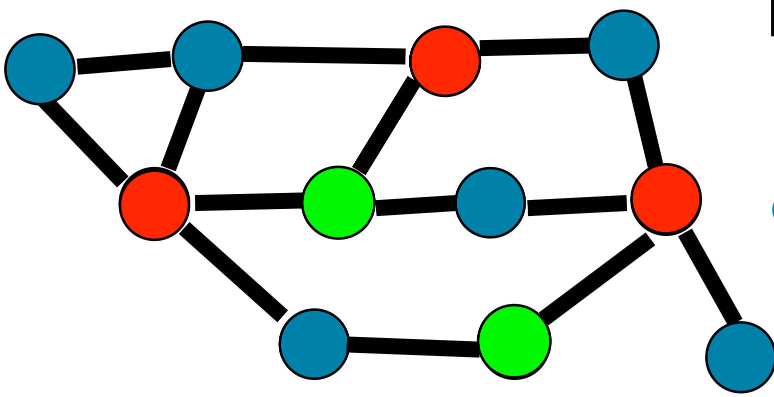
AMG coarsening and interpolation

- *large a_{ij}, 'strong connections' are important*
- *define strength matrix S:*

$$A = \begin{bmatrix} x & x & x & & \\ & x & x & & x \\ & x & x & x & x \\ & x & x & x & \\ x & & & x & x \end{bmatrix} \qquad S = \begin{bmatrix} 1 & 1 & 0 & & \\ & 1 & 0 & & 1 \\ & 0 & 0 & 1 & 1 \\ & 1 & 0 & 1 & \\ 1 & & & 1 & 0 \end{bmatrix}$$

- *consider the undirected graph of S*
- *apply parallel maximal independent set algorithm to graph(S) [Luby, 1986]*

classical AMG coarsening (CLJP)



- (C1) Maximal Independent Set:

Independent: no two C-points are connected

Maximal: if one more C-point is added, the independence is lost

- (C2) All F-F connections require connections to a common C-point (for good interpolation)
- F-points have to be changed into C-points, to ensure (C2); (C1) is violated

more C-points, higher complexity

classical coarsening: scalability results

- *example: finite difference Laplacian, parallel CLJP coarsening algorithm*

- *2D (5-point): near-optimal scalability* **(250^2 dof/proc)**

Procs	C_{op}	t_{tot}	Iter
16	4.48	2.89	9
64	4.50	3.85	9
256	4.50	5.01	9

(2) classical coarsening may lead to complexity problems

- *3D (7-point): complexity growth*

dof	C_{op}
32^3	16.17
64^3	22.51

(3) Parallel Modified Independent Set (PMIS) coarsening

our approach to reduce complexity:

- *do not add C points for strong F-F connections that do not have a common C point*

- *less C points, reduced complexity, but worse convergence factors expected*

- *compensate by GMRES acceleration*

parallel PMIS results: 7-point finite difference Laplacian in 3D, 40^3 dof per proc

CLJP and *PMIS-GMRES(10)*

proc	C_{op}	Iter	t_{total}
1	14.39	6	3.35
512	17.02	10	35.83
1331	17.19	10	46.25
1	*2.32*	*13*	*1.28*
512	*2.37*	*25*	*12.77*
1331	*2.37*	*28*	*17.99*

convergence problems on PMIS-coarsened grids

- *PMIS coarsening works well for many problems, but requires GMRES acceleration*
- *for some problems, too many iterations are necessary because interpolation is not accurate enough ("not enough C-points")*
- *one solution: add C-points (CLJP…)*
- *other solution: use distance-two C-points for interpolation = long-range interpolation*

  ➔ *F-F interpolation*

convergence problems

- *3D elliptic PDE with jumps in coefficient a*

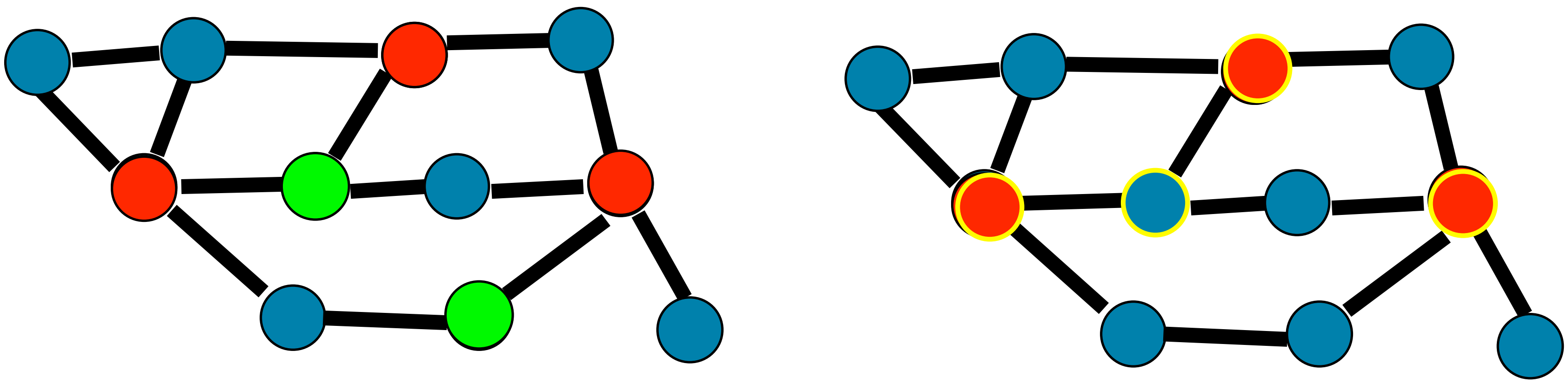
$$(au_x)_x + (au_y)_y + (au_z)_z = 1$$

- *1000 processors, 40^3 dof/proc*

	t_{tot}	C_{op}	Iter
CLJP	52.48	17.00	17
PMIS	211.79	2.40	686

- *remedy: improve interpolation used with PMIS*

(4) improving interpolation: F-F interpolation



- *when strong F-F connection without a common C-point is detected, do not add C-point, but extend interpolation stencil to distance-two C-points*
- *no C-points added, but larger interpolation stencils*

results using long-range F-F interpolation

- *3D elliptic PDE with jumps in coefficient a*

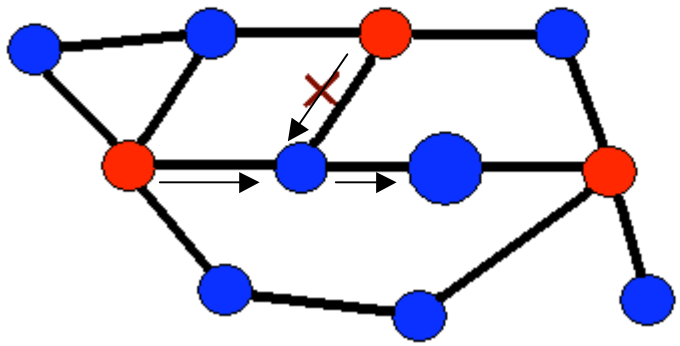
$$(au_x)_x + (au_y)_y + (au_z)_z = 1$$

- *1 processor, AMG+GMRES, 80^3 dof*

	t_{tot}	C_{op}	Iter
CLJP	48.0	21.54	7
PMIS	94.6	2.46	188
PMIS + F-F	21.4	4.90	9

reduce complexity: FF1 Interpolation

- *Modified FF Interpolation (FF1)*



- *To reduce operator complexity, only include one distance-two C-point when a strong FF connection is encountered*
- *Setup time, complexity are reduced*

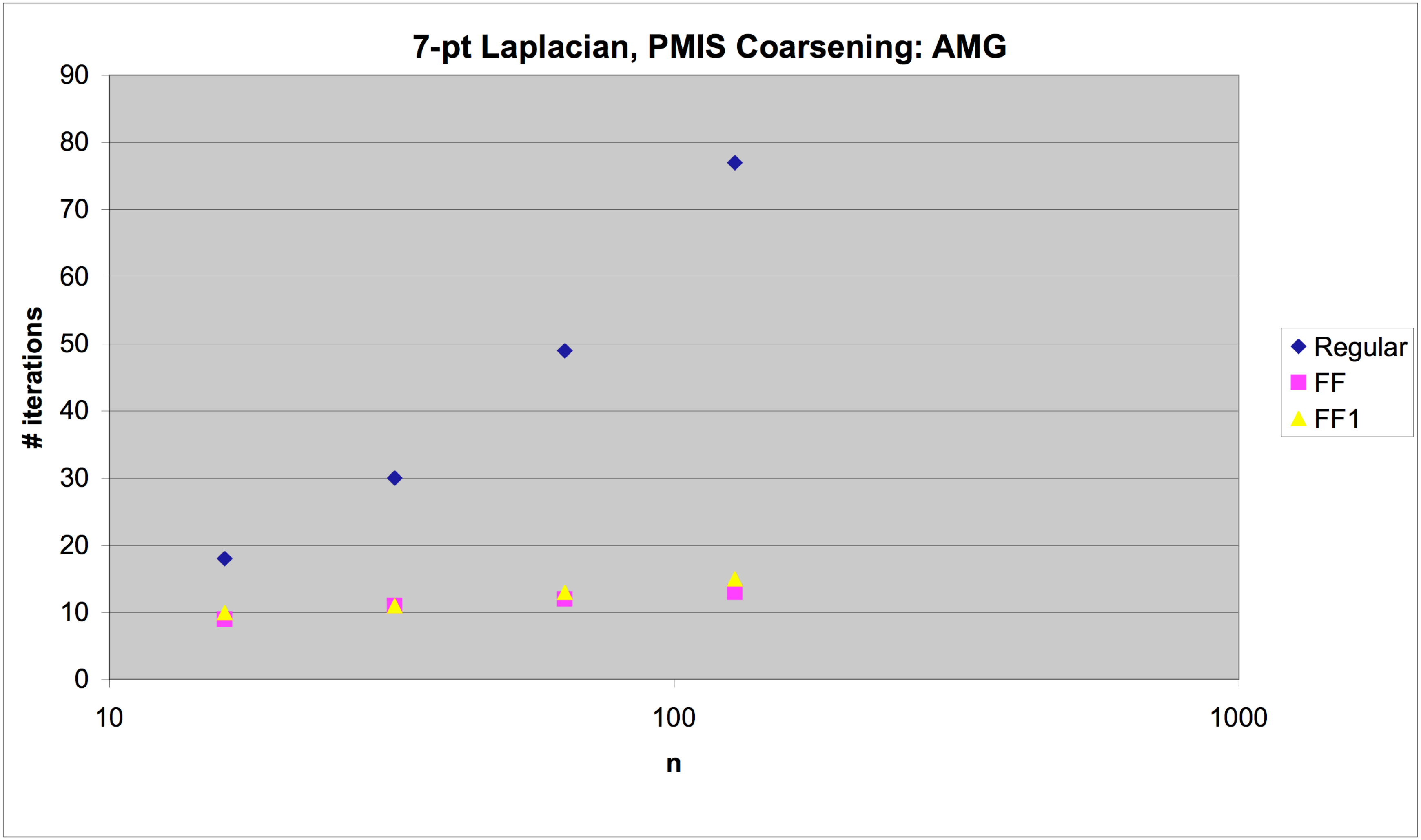
results: 7-pt Laplacian Problem

- **PMIS coarsening, 1 processor, 128^3 dof**

- **AMG**

	C_{op}	# iterations	t_{setup} (s)	t_{solve} (s)	t_{total} (s)
Regular	2.36	77	16.63	85.93	102.56
FF	4.80	13	83.81	22.86	106.67
FF1	3.68	15	44.22	22.07	66.29

scalability of PMIS-FF1



results: 3D elliptic PDE with jumps

$$(au_x)_x + (au_y)_y + (au_z)_z = 1$$

- **AMG, 1 processor, 120^3 dof**

	C_{op}	# iterations	t_{setup} (s)	t_{solve} (s)	t_{total} (s)
Regular	2.44	>> 200	*Slow to*	*converg e*	
FF	4.94	14	62.95	20.54	83.49
FF1	3.84	18	35.36	22.47	57.83

(5) scaling results on Blue Gene/L

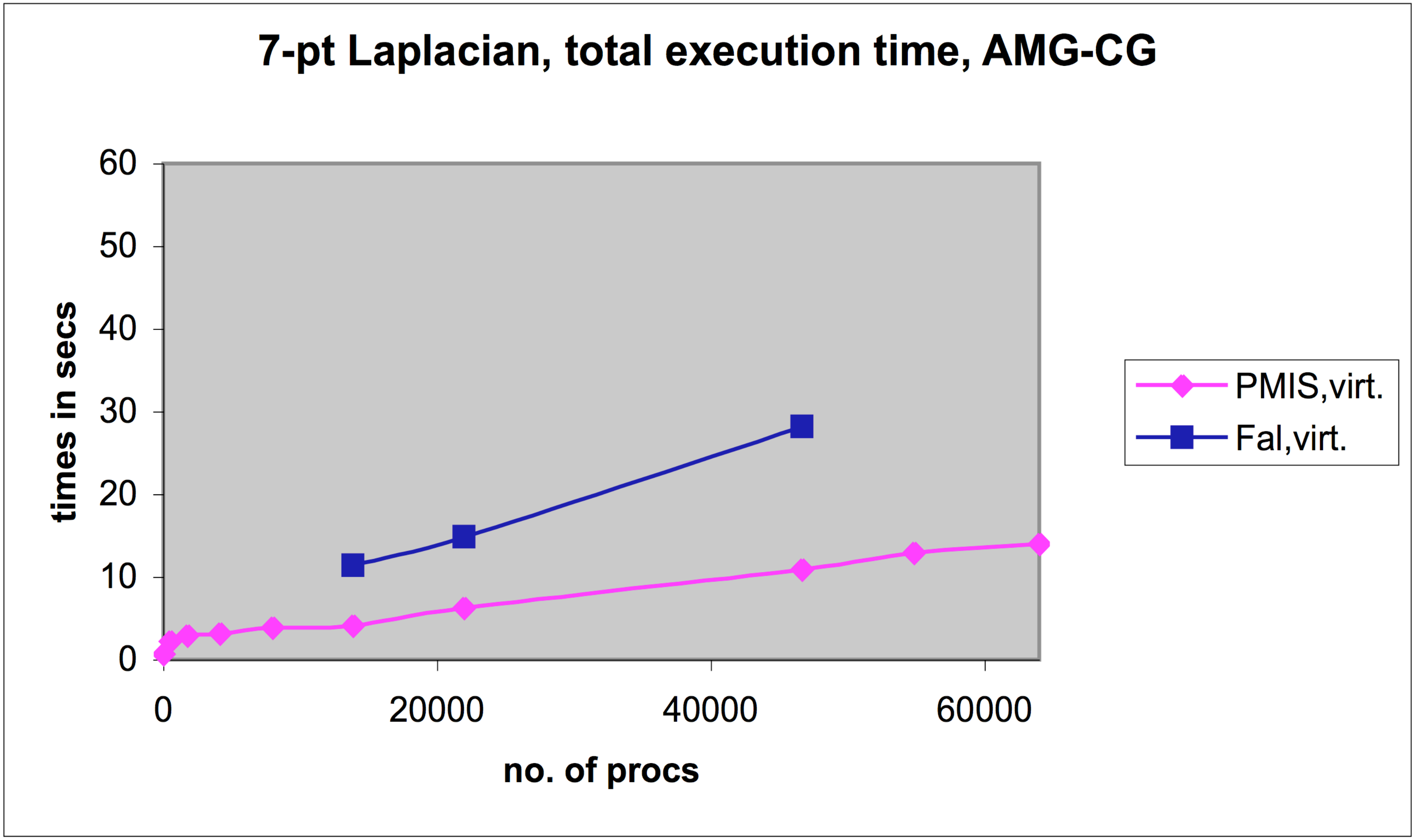
Top 500 Supercomputer list (November 2005)

Rank	Site	System	Processors	Rmax (Gflop)
1	Livermore Lab, US	IBM Blue Gene/L	131,072	280,600
2	Thomas J. Watson, US	IBM Blue Gene	40,960	91,290
3	Livermore Lab, US	IBM pSeries	10,240	63,390
4	NASA/Ames, US	SGI Altix	10,160	51,870
5	Sandia Lab, US	Dell PowerEdge	8,000	38,270
6	Sandia Lab, US	Cray XT3	10,880	36,190
7	Earth Simulator Center, Japan	Earth Simulator NEC	5,120	35,860
8	Barcelona Supercomputer Center, Spain	IBM JS20 Cluster	4,800	27,910
9	University Groningen, Netherlands	IBM Blue Gene	12,288	27,450
10	Oak Ridge Lab, US	Cray XT3	5,200	20,527

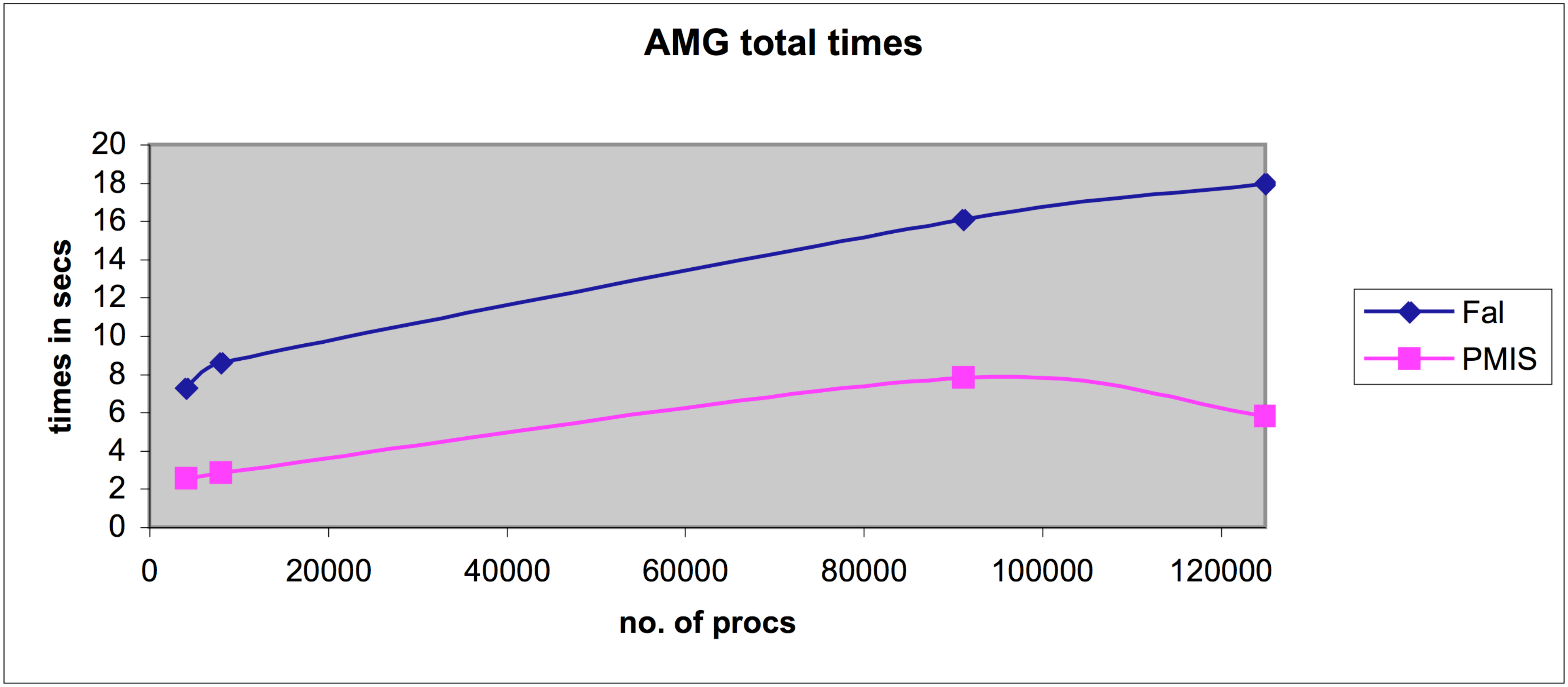
LLNL Blue Gene/L

- *dual-processor nodes optimized for data access*
- *each node: one processor for simulation, one for communication; only 256MB ram per processor*
- *lightweight, single-process linux kernel*

LLNL Blue Gene/L results

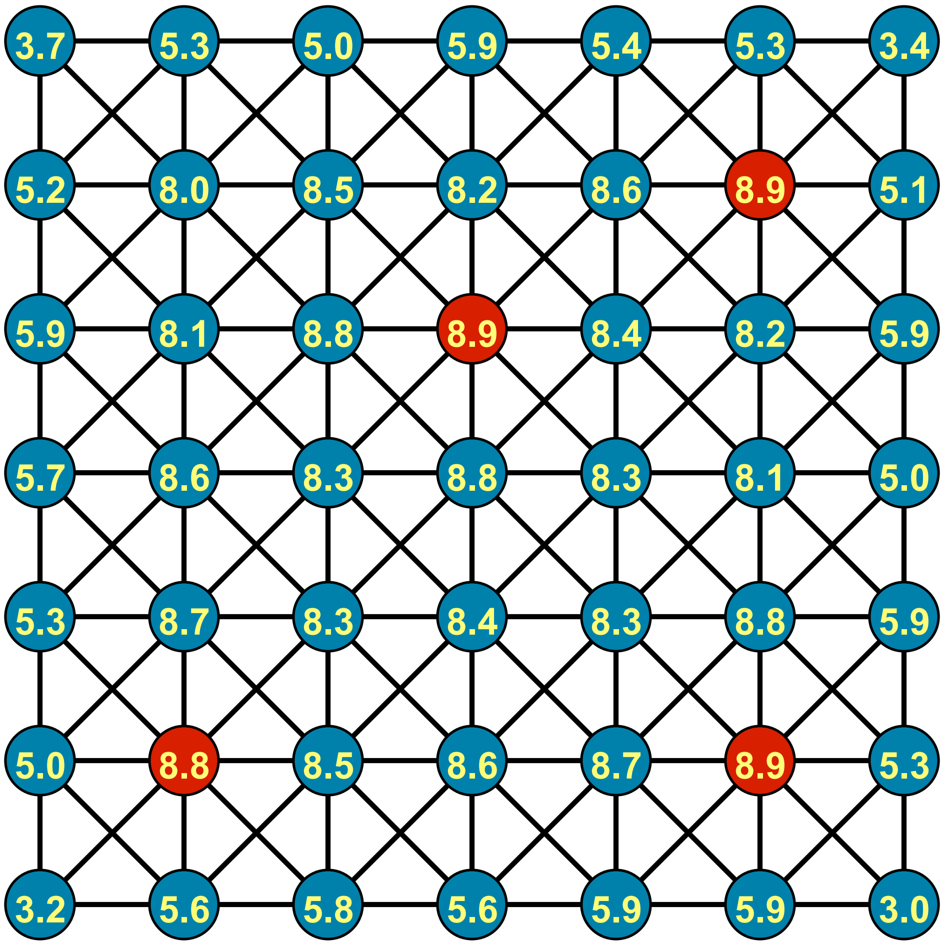


LLNL Blue Gene/L results on full machine



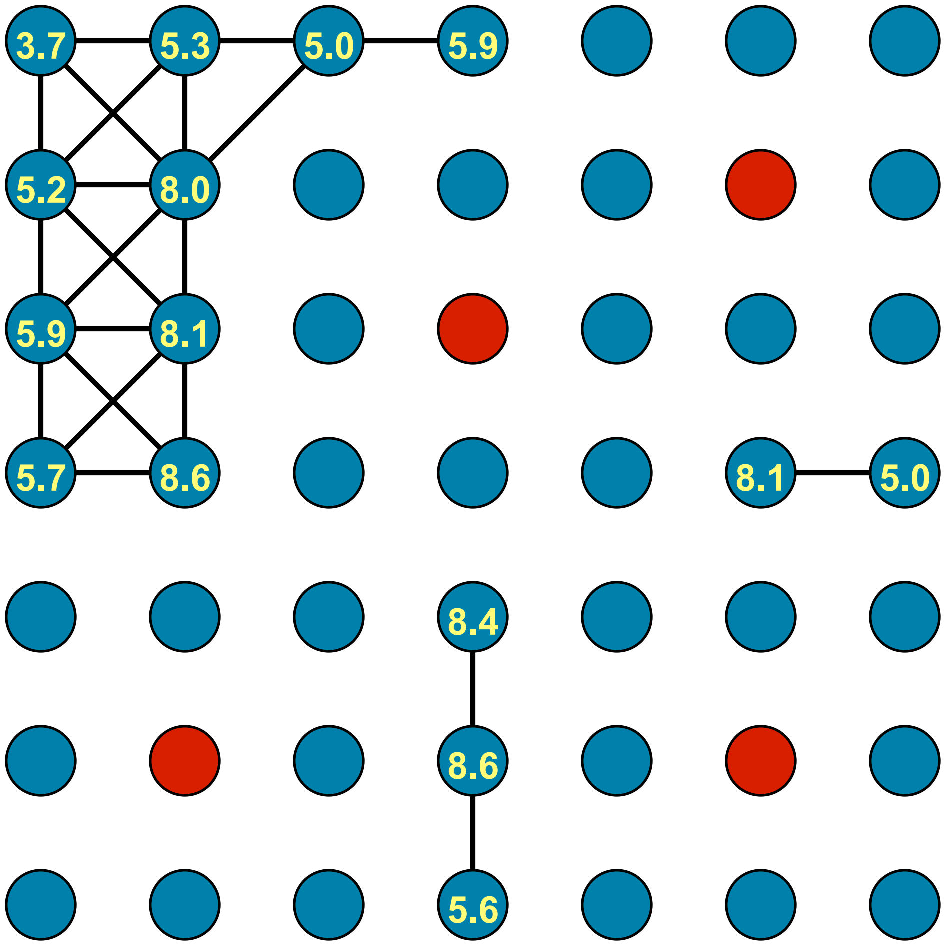
7-pt Laplacian, total execution time, AMG-CG, total problem size ~2 billion

PMIS: select 1



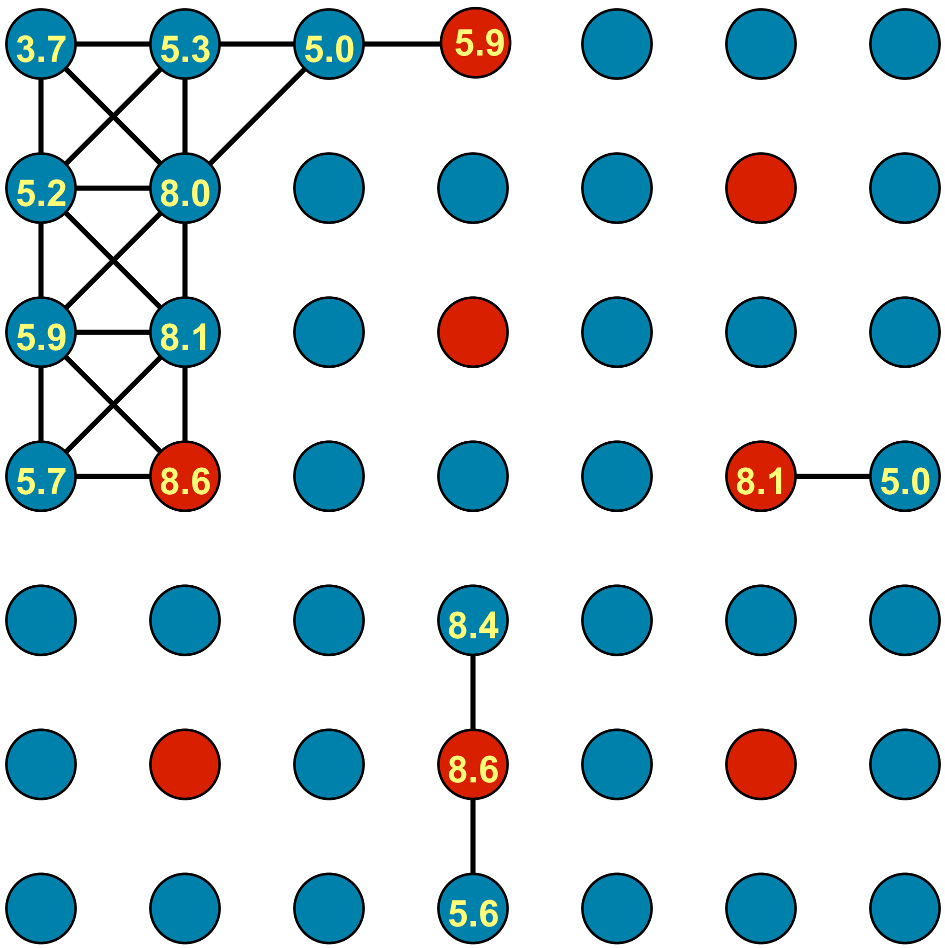
- select C-pts with maximal measure locally
- make neighbour F-pts
- remove neighbour edges

PMIS: remove and update 1



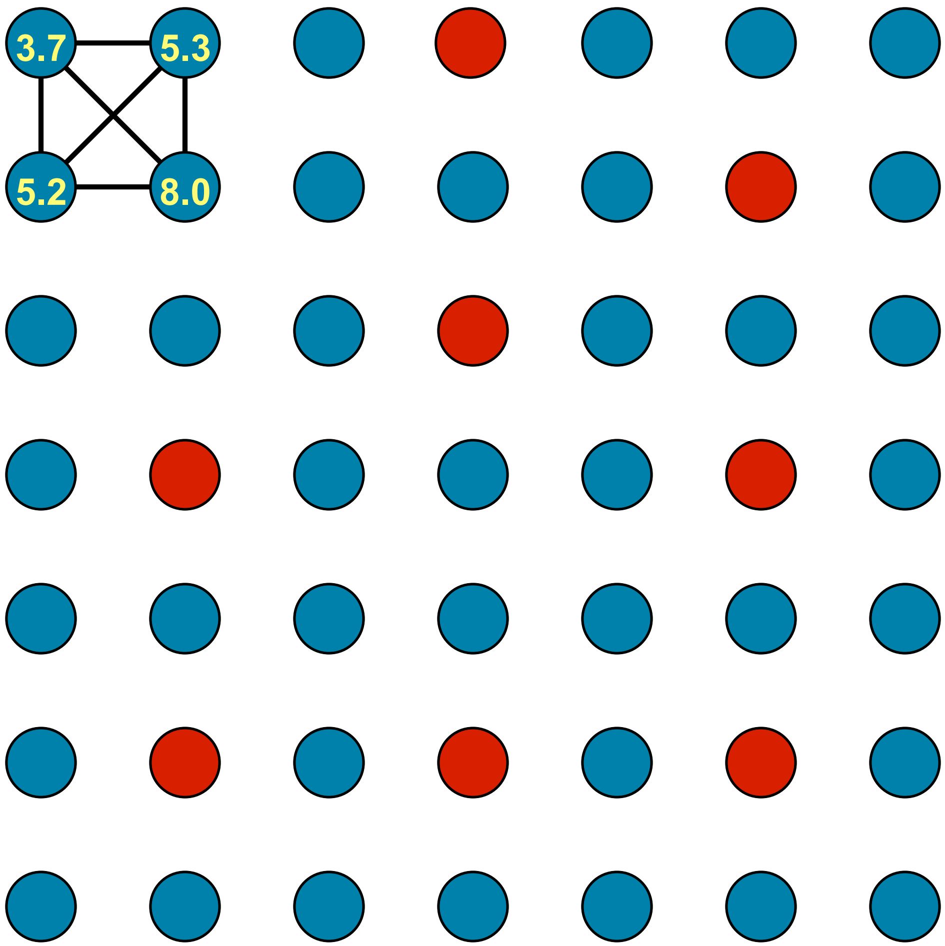
- select C-pts with maximal measure locally
- make neighbours F-pts
- remove neighbour edges

PMIS: select 2



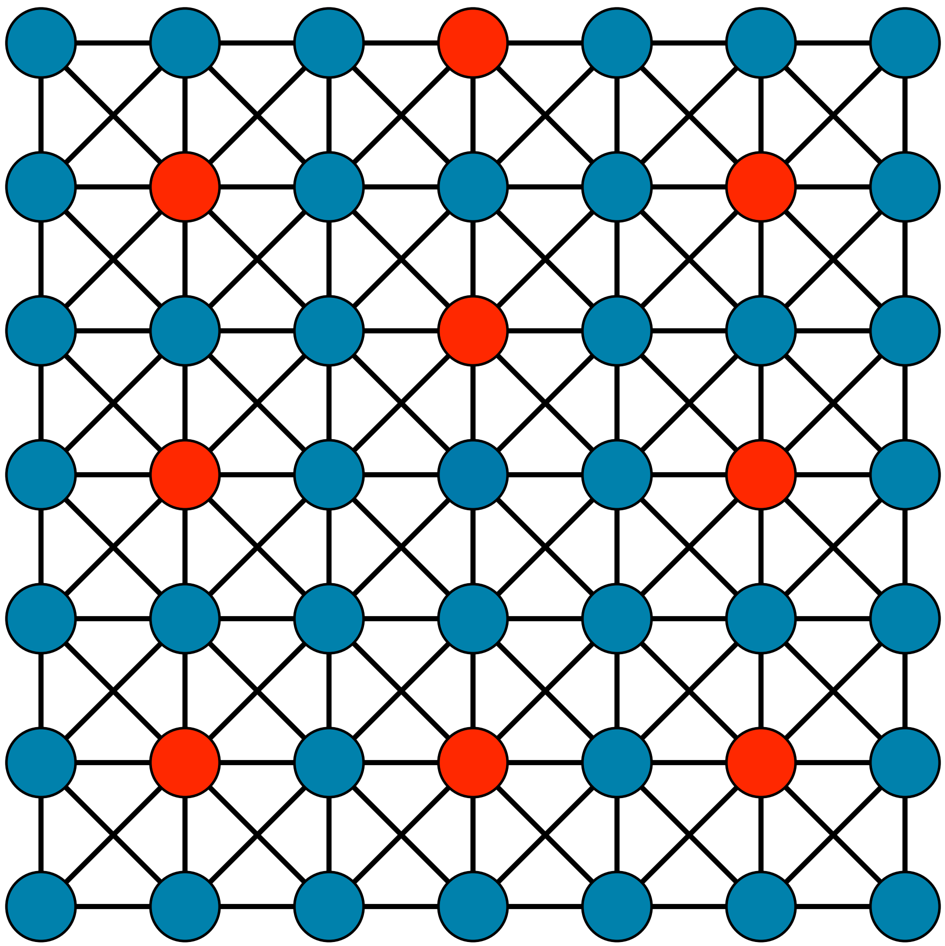
- select C-pts with maximal measure locally
- make neighbours F-pts
- remove neighbour edges

PMIS: remove and update 2



- select C-pts with maximal measure locally
- make neighbours F-pts
- remove neighbour edges

PMIS: final grid



- select C-pts with maximal measure locally
- make neighbour F-pts
- remove neighbour edges
- parallel algorithm

LLNL Blue Gene/L results