

# *Towards a Scalable Parallel Sparse Linear System Solver*

*M. Manguoglu*

*METU, Turkey*

*A. Sameh\**

*Purdue University, U.S.A.*

*O. Schenk & M. Sathe*

*Univ. of Basel, Switzerland*

*Acknowledge: F. Saied (CS, Purdue)*

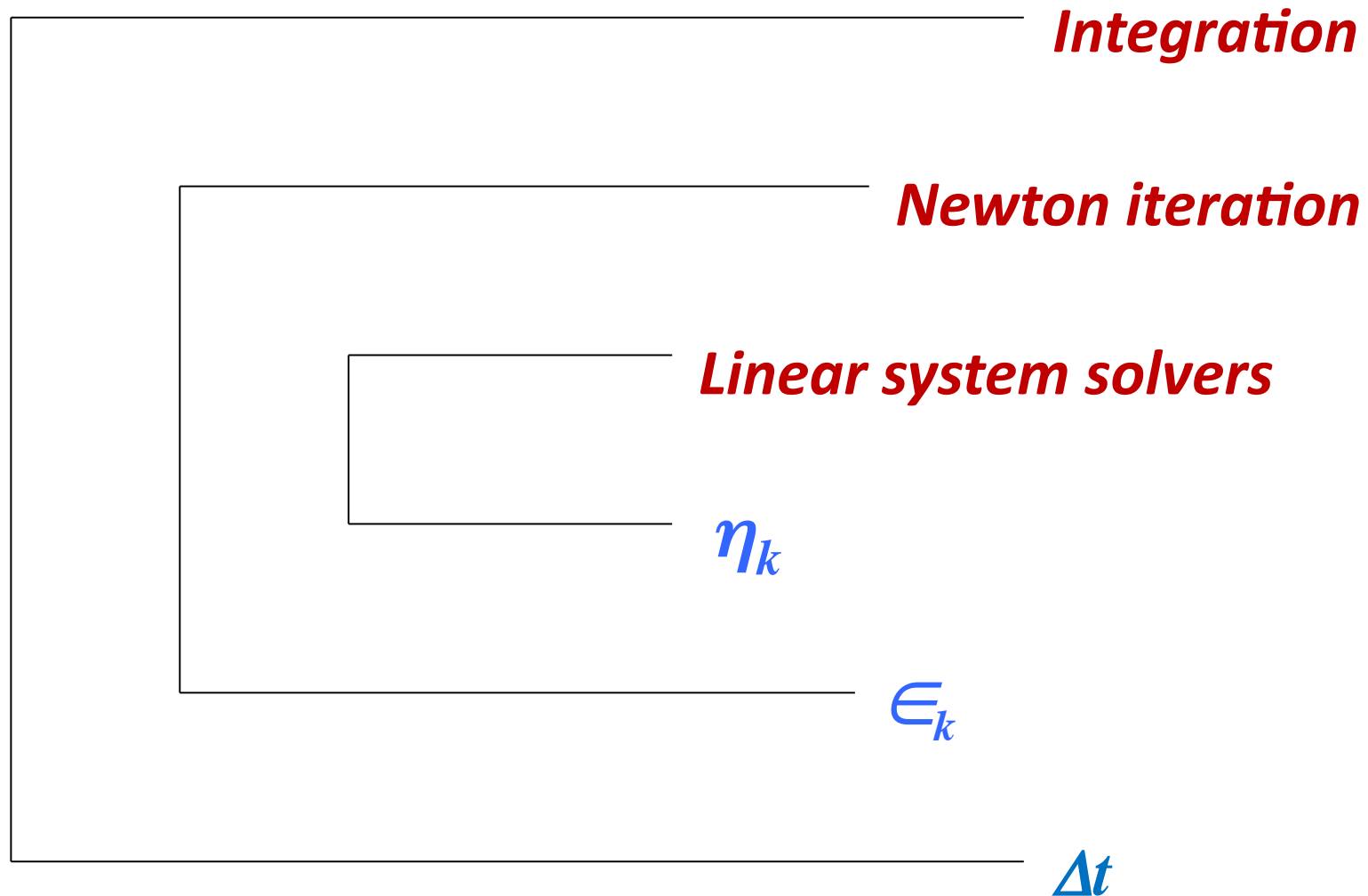
*Support\*: ARO, Intel, NSF.*

*Sparse Days 2011  
CERFACS, September 6 & 7*

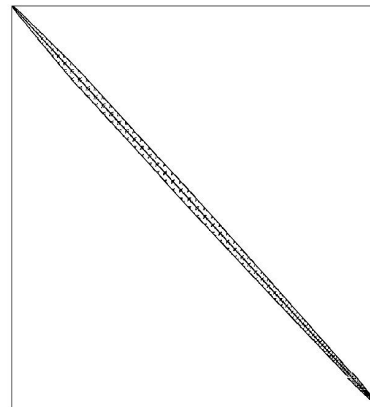
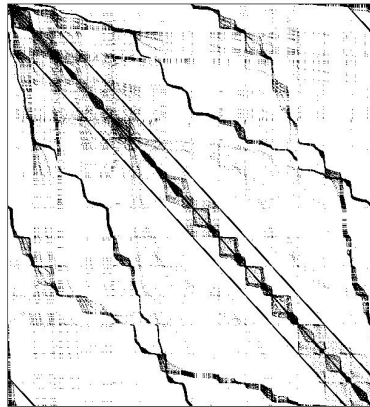
# Outline

- *Motivating applications – a sample*
- *The parallel scalability of the SPIKE family of banded linear system solvers*
- *Extensions for solving general sparse linear systems:*
  - *The Pardiso-SPIKE hybrid solver (PSPIKE)*
- *Role of reordering for faster MATVEC and extraction of preconditioners:*
  - *TraceMIN – a parallel eigensolver for computing the Fiedler vector (spectral reordering)*

# *Target Computational Loop*



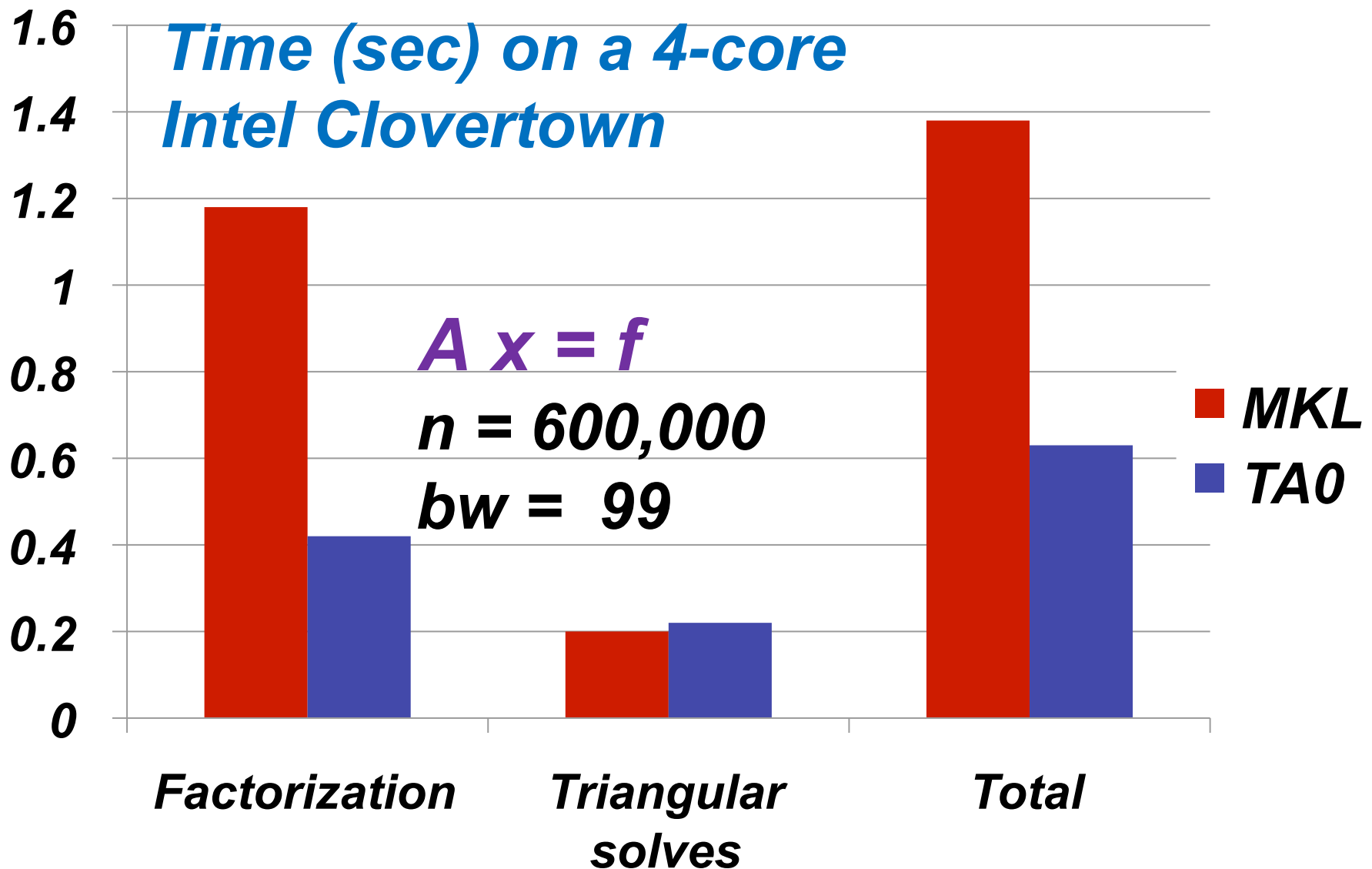
# Role of Reordering Schemes

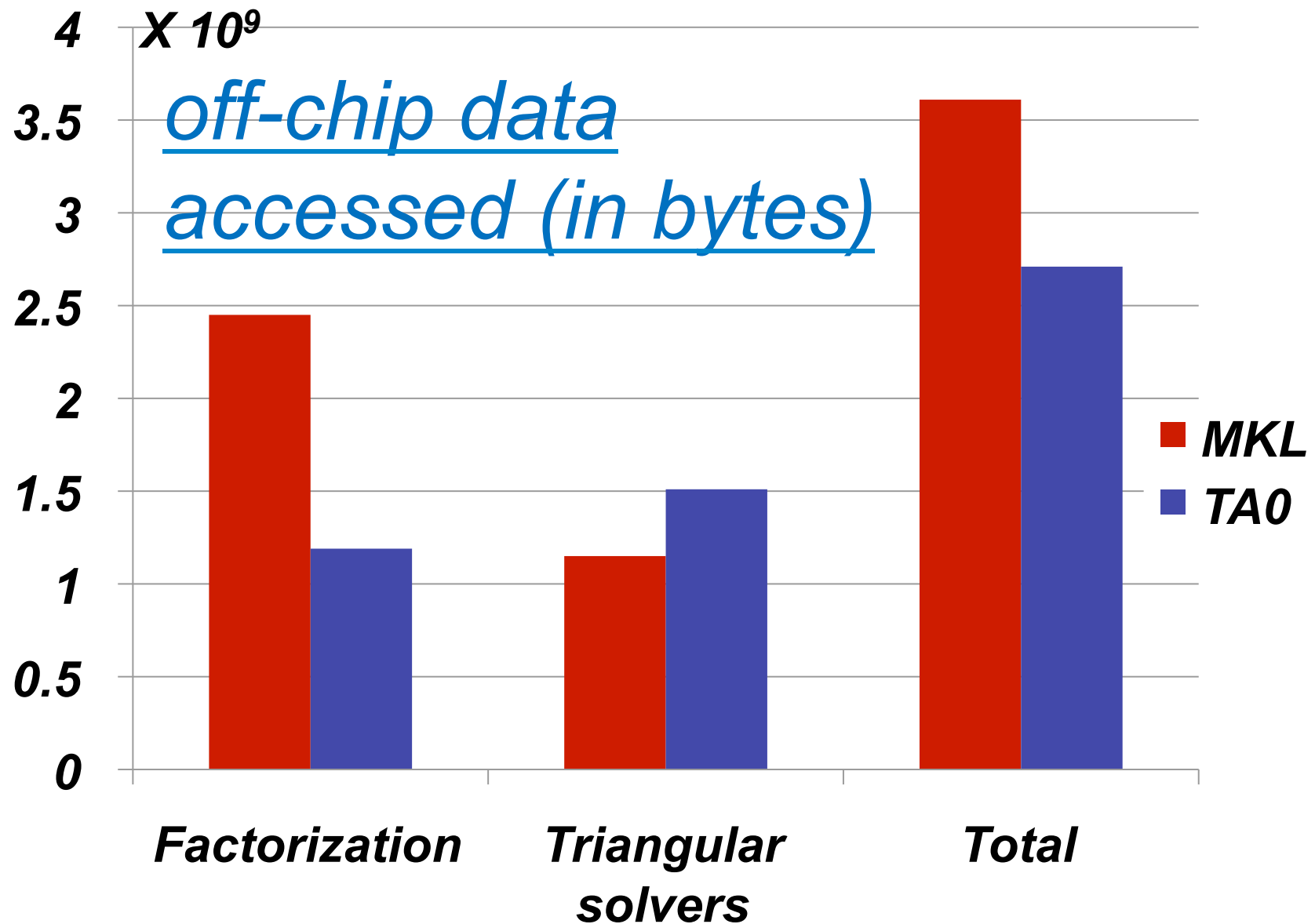


Banded  
Preconditioners

*after RCM reordering*

- “Banded”, or low-rank perturbations of banded, systems are often obtained after *RCM* or *spectral reordering*,
- This yields more scalable parallel matrix-vector multiplications, and helps in extracting more effective “banded” preconditioners.





*“Analyzing memory access intensity in parallel programs for multicore architectures”*

*L. Liu, Z. Li, and A. S.*

# *Spike-based sparse solver: **PSPIKE***

*Step 1: reordering (equivalent to HSL -- MC64 + MC73)*

*Step 2: extract a “banded” preconditioner  $M$*

*BiCGstab*

*(or any Krylov  
subspace method)*

*Solve  $Mz = r$  (via Pardiso-SPIKE)*

Manguoglu M, Sameh A, and Schenk O, PSPIKE: A Parallel Hybrid Sparse Linear System Solver, Lecture Notes in Computer Science(proceedings of EuroPar09), Volume 5704, pp.797-808, 2009

# *Weighted Spectral Reordering (WSO)*

## *Main Objectives:*

- encapsulating as many of the heaviest sparse matrix elements as possible within a central band **to be used as a preconditioner**,*
  - reducing the rank of the matrix lying outside the central band.*
- realizing a faster “matvec”*

*(banded matrix + few nonzero elements outside the band)*

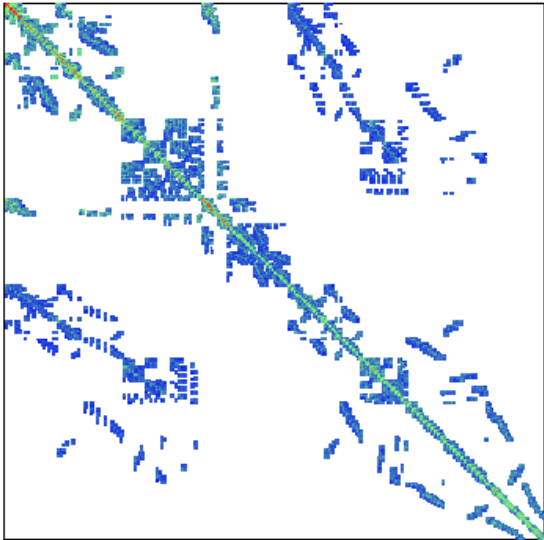


# *UFL: smt -- structural mechanics*

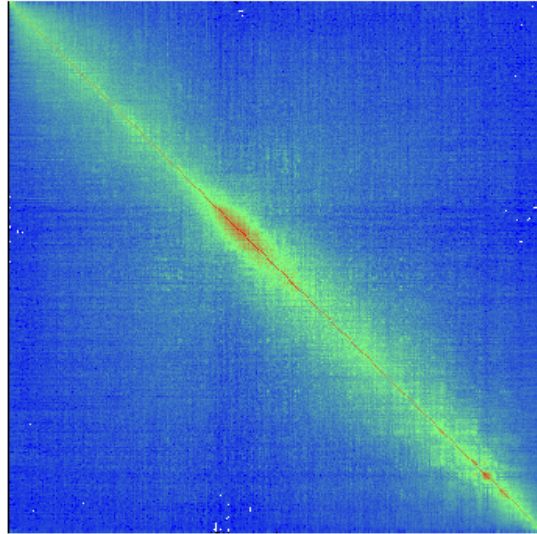
N: 25,710 NNZ: 3,749,582

*after HSL-MC73*

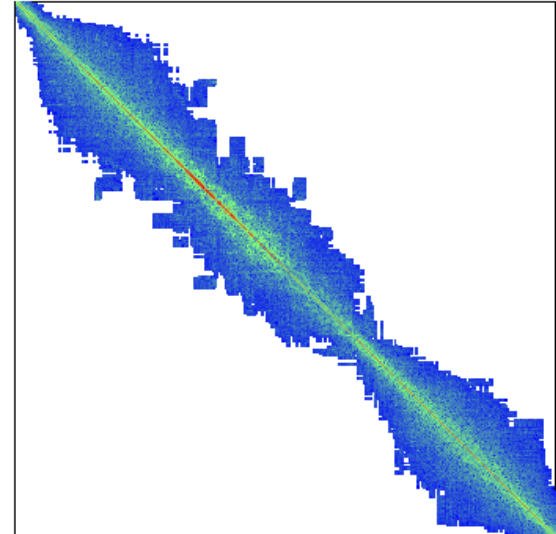
*after*  
*TraceMIN-Fiedler*



Original matrix



After MC73

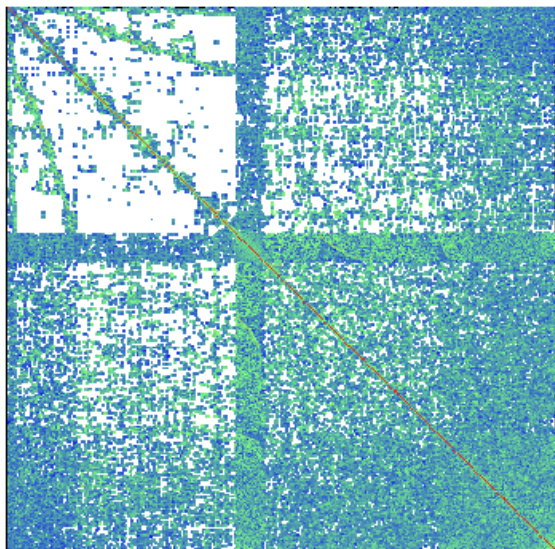


After TraceMin-Fiedler

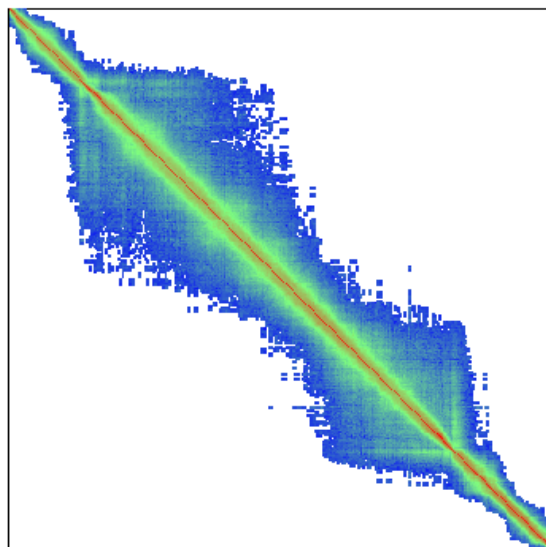
*obtaining the Fiedler vector via the eigensolver: TraceMIN*  
*(Wisniewski and A.S. -- SINUM, '82)*

# *UFL: f2 -- structural mechanics*

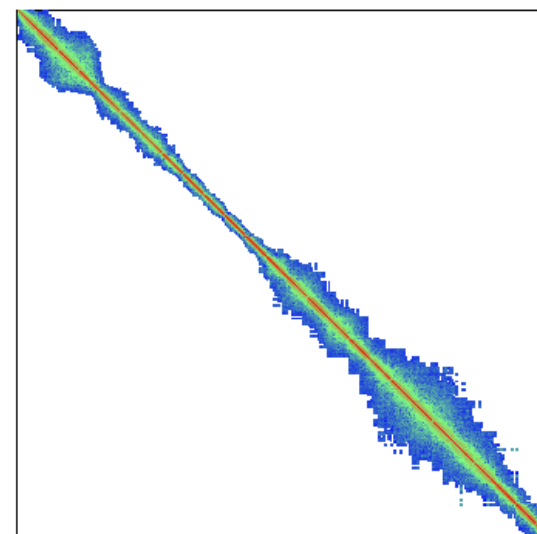
*N: 71,505 NNZ: 5,294,285*



Original matrix



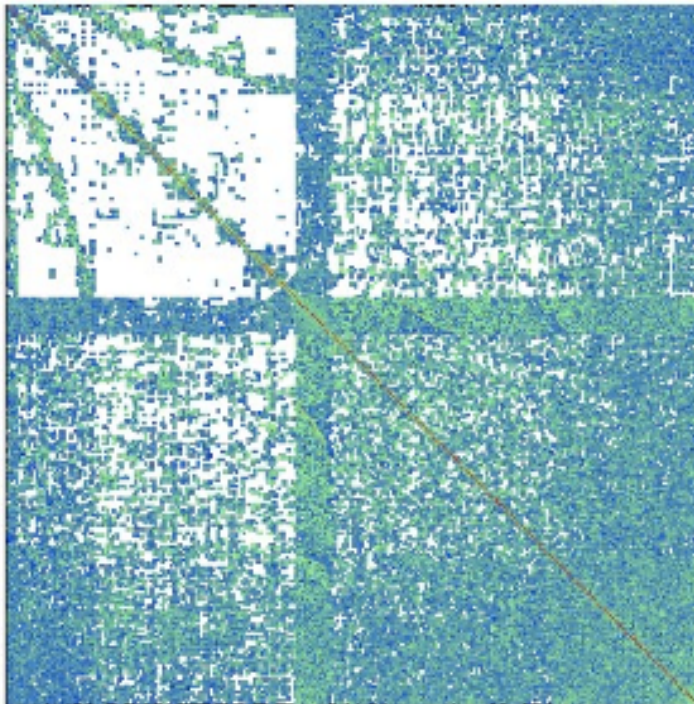
After MC73



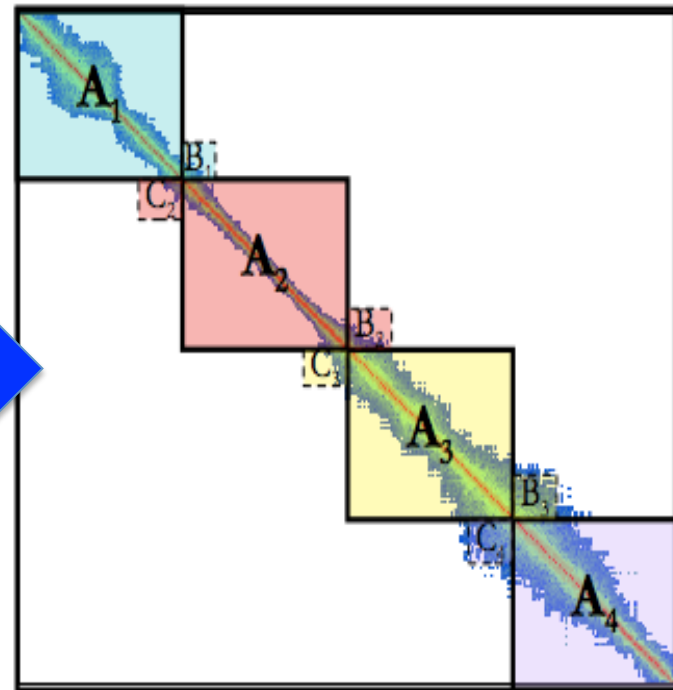
After TraceMin-Fiedler

*TraceMIN-Fiedler: Murat Manguoglu et. al -- submitted*

# $UFL - f2$

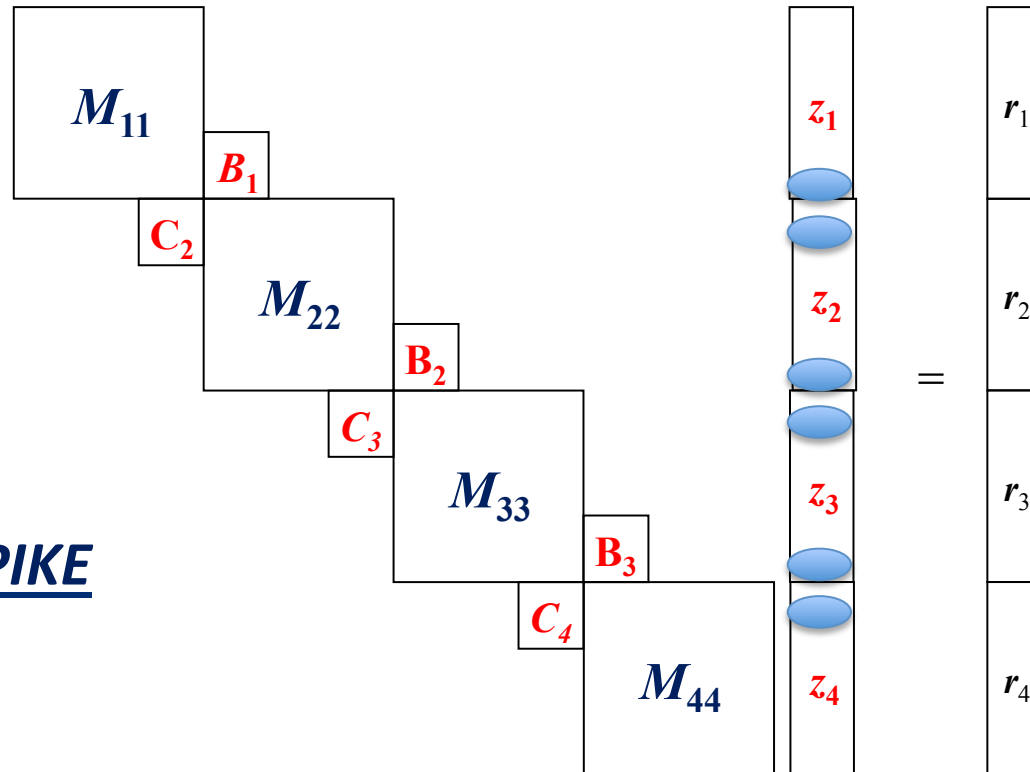


*Before reordering*



*After reordering  
via TraceMIN-Fiedler*

$$M z = r \quad (M \text{ is "banded"})$$



Each  $M_{kk}$  is a  
general sparse  
matrix

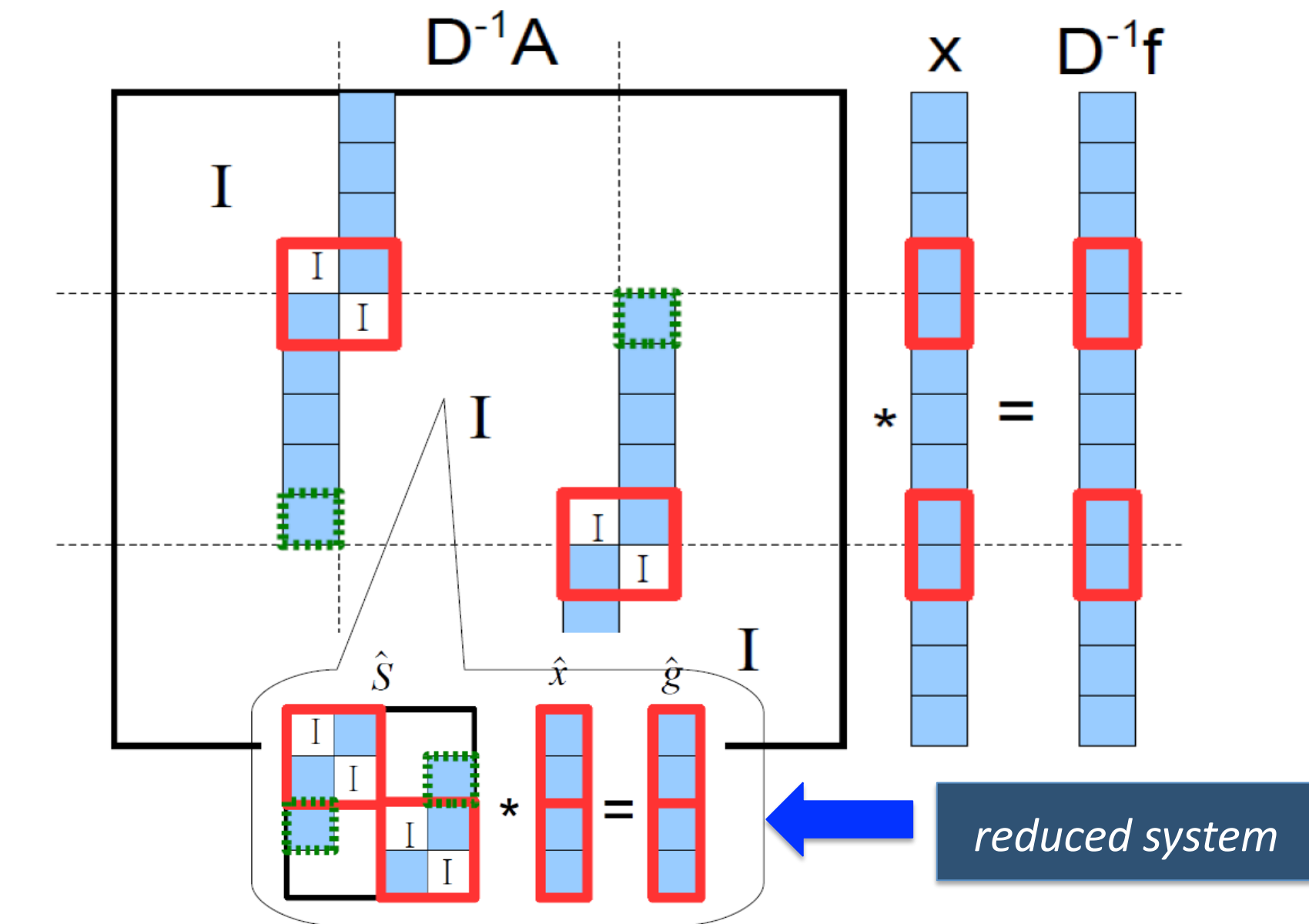
**PSPIKE:**  
Pardiso-SPIKE

$$P = M + \delta(M) = D' * S'$$

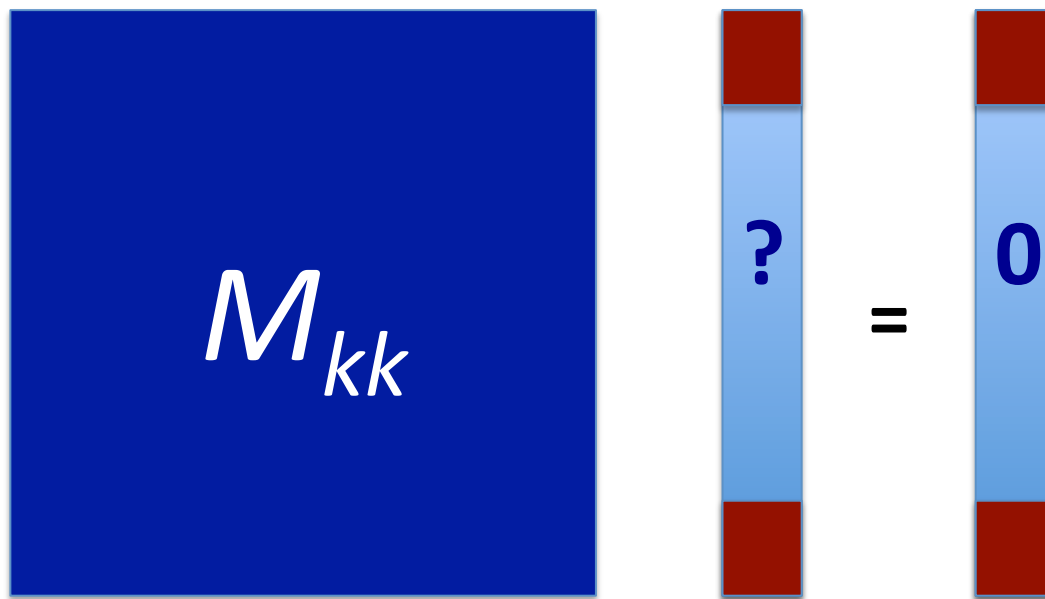
(i) Solve  $D' y = r$

(ii) Solve  $S' z = y$

**Solving systems involving  
The preconditioner  $P z = r$**

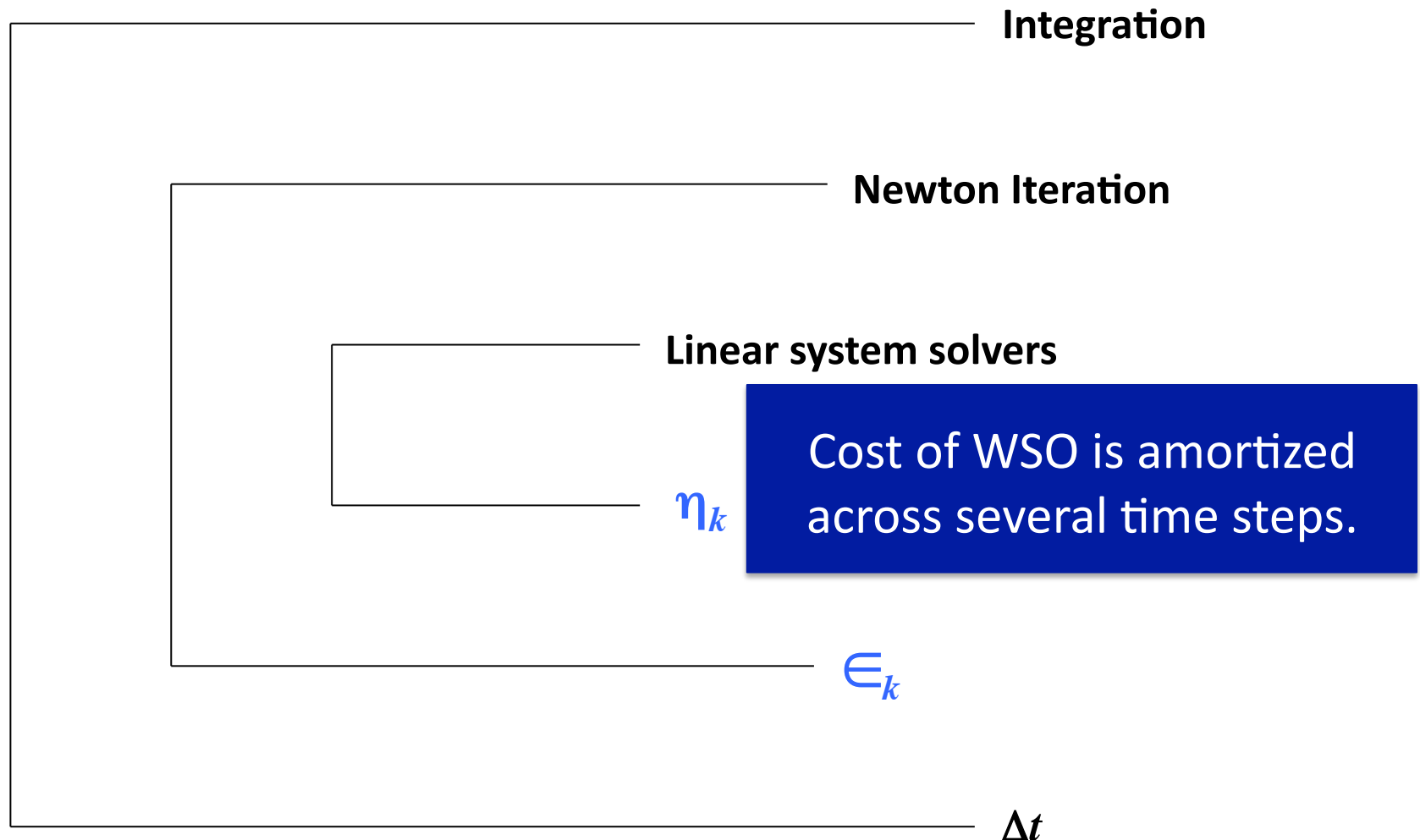


## *Generating tips of the spikes*



*Obtain the upper and lower tips of the solution block via the **modified** direct sparse system solver “**Pardiso**”.*

# *How expensive is spectral reordering?*



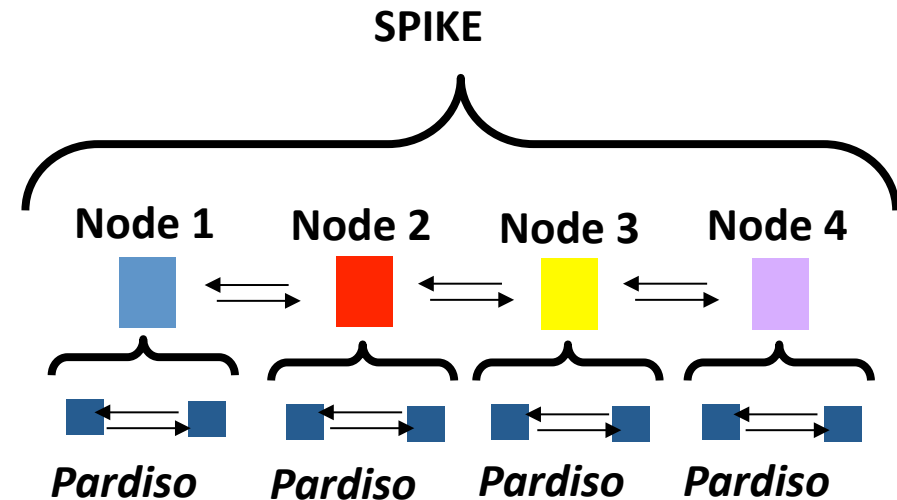
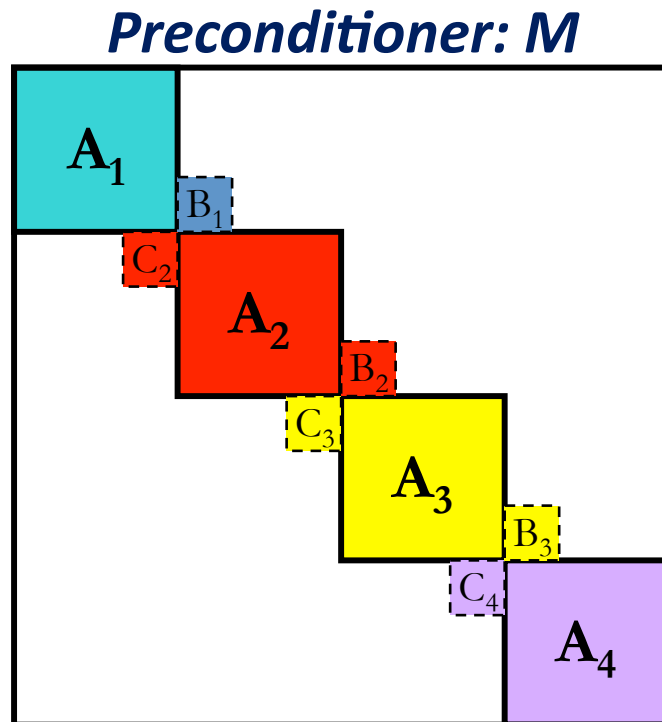
will return to this issue later

$\sigma = T(\text{MC73}) \text{ on } 1 \text{ core} \div$   
 $T(\text{TraceMIN-Fiedler}) \text{ on } 4 \text{ nodes}$

<i>Matrix</i>	<i>n</i>	<i>nnz</i>	<i><math>\sigma</math></i>
<i>Rajat31</i>	<i>~4.7 M</i>	<i>~ 20 M</i>	<i>85</i>
<i>Schenk/nlpkkt20</i>	<i>~ 3.5 M</i>	<i>~ 95 M</i>	<i>8</i>
<i>Freescale1</i>	<i>~ 3.4 M</i>	<i>~ 17 M</i>	<i>4</i>
<i>KKT-power</i>	<i>~ 2.1 M</i>	<i>~ 13 M</i>	<i>641</i>



# Multilevel Parallelism of PSPIKE for solving $Mz = r$



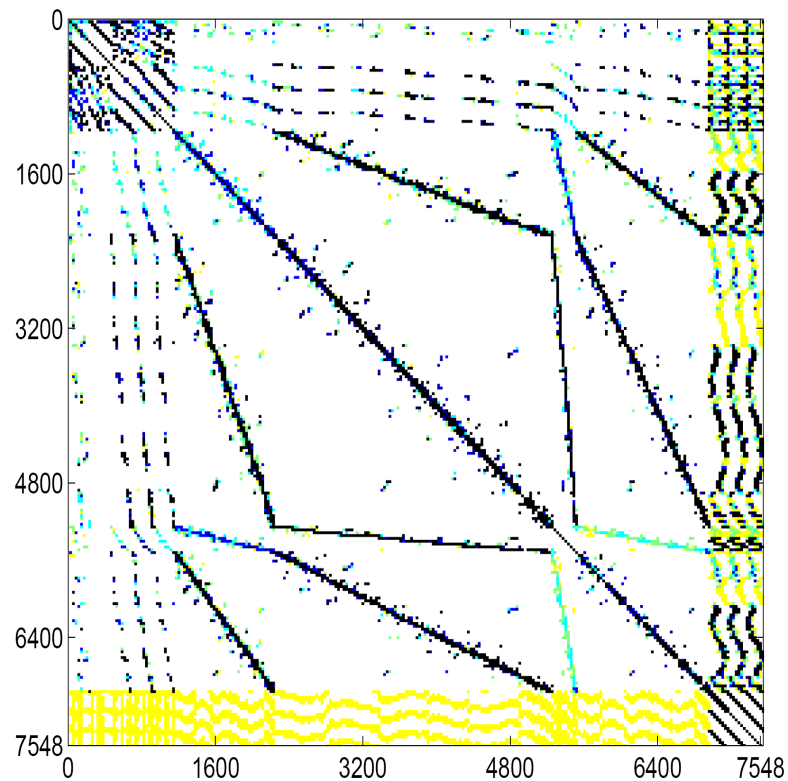
*Preconditioner  $M$  is generalized-banded (MBP)*

*Other cases:*

- *narrow-banded (NBP)*
- *wide-banded (WBP)*

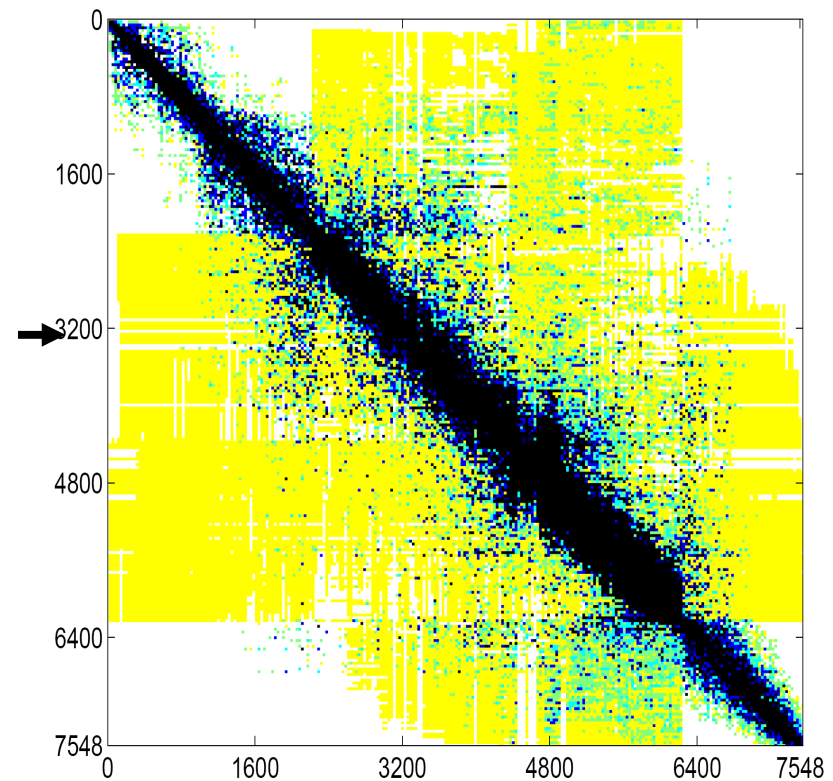
# *Circuit Simulation*

Original Matrix



*after TraceMIN-Fiedler*

Reordered Matrix

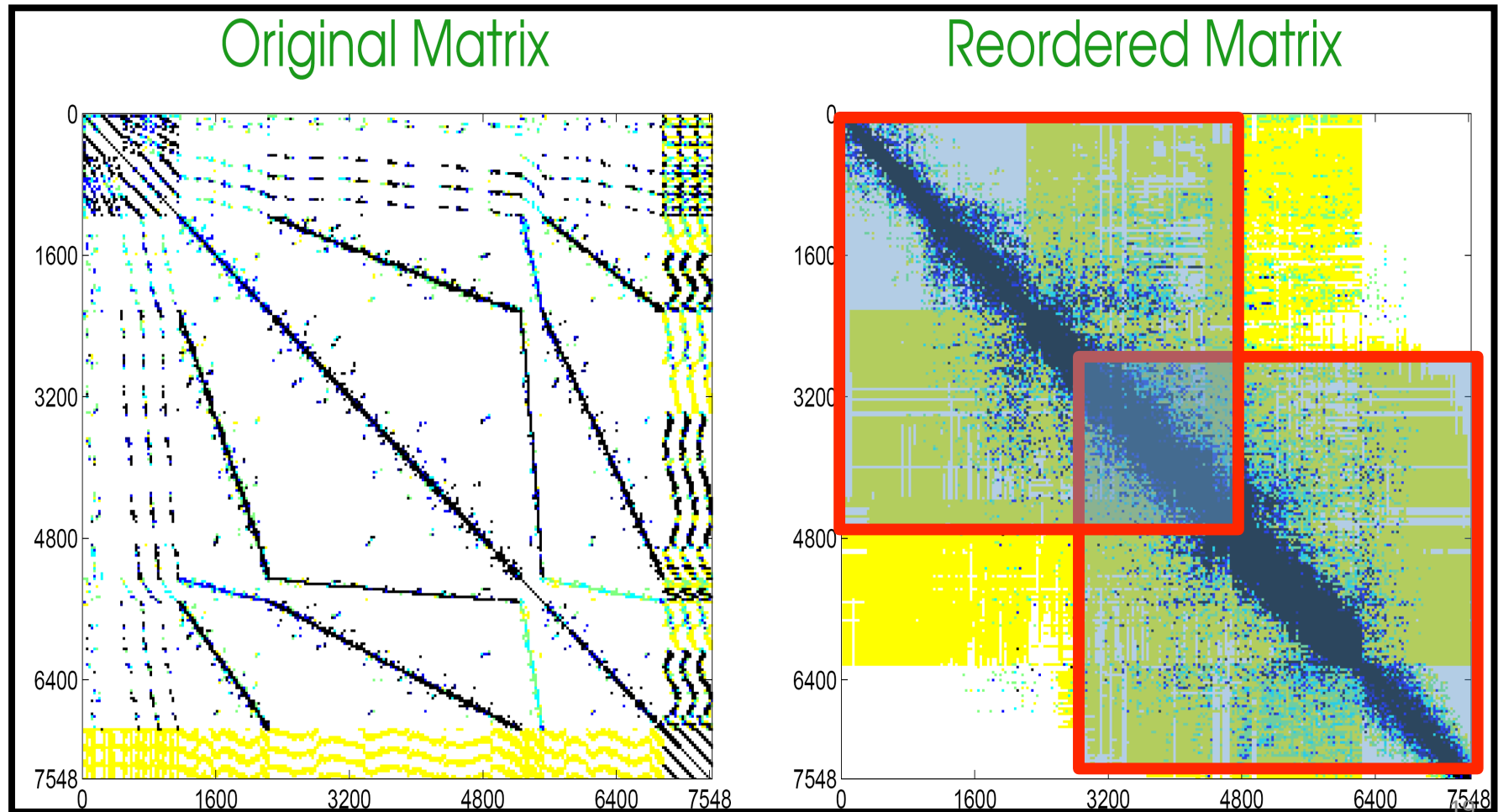


*Wide-Banded Preconditioner*

# WBP: wide-banded preconditioners

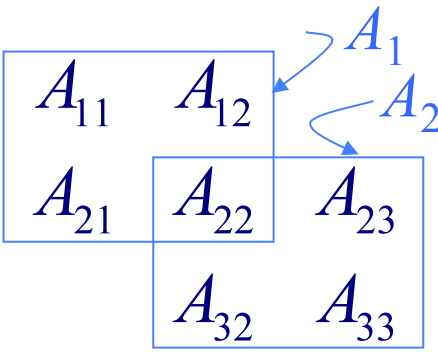
*preconditioner consists of overlapped blocks*

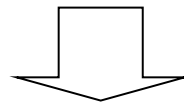
*Naumov, Manguoglu, A.S. : A Tearing-based Hybrid Parallel Sparse Linear System Solver: JACM, 2010.*



Solve:  $Ax = f$  ;  $A := \text{nonsingular}$

$$\begin{bmatrix} \boxed{\begin{matrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{matrix}} & \begin{matrix} A_{12} \\ A_{23} \\ A_{32} & A_{33} \end{matrix} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}$$





$$(n \times n) \begin{bmatrix} A_{11}^{(1)} & A_{12}^{(1)} \\ A_{21}^{(1)} & A_{22}^{(1)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} f_1 \\ \alpha f_2 + y \end{bmatrix} ; 0 < \alpha < 1$$

$$(n \times n) \begin{bmatrix} A_{11}^{(2)} & A_{12}^{(2)} \\ A_{21}^{(2)} & A_{22}^{(2)} \end{bmatrix} \begin{bmatrix} \hat{x}_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} (1 - \alpha) f_2 - y \\ f_3 \end{bmatrix}$$

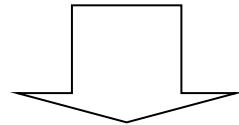
$$A_{22} = A_{22}^{(1)} + A_{11}^{(2)} \in \mathfrak{R}^{m \times m}$$

$$m \ll n$$

$$A_1 \rightarrow \begin{bmatrix} A_{11}^{(1)} & A_{12}^{(1)} \\ A_{21}^{(1)} & A_{22}^{(1)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} f_1 \\ \alpha f_2 + y \end{bmatrix}$$

$$A_2 \rightarrow \begin{bmatrix} A_{11}^{(2)} & A_{12}^{(2)} \\ A_{21}^{(2)} & A_{22}^{(2)} \end{bmatrix} \begin{bmatrix} \hat{x}_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} (1-\alpha)f_2 - y \\ f_3 \end{bmatrix}$$

Determine that  $\dot{y}$  which assures that  $x_2 = \hat{x}_2$ .



$\dot{y}$  is the solution of the *balance system* (order = size of overlap)

$$M y = g$$

*M and g are not available explicitly*

# *The Balance System*

- *Solve the system  $My=g$  using a Krylov subspace scheme (CG or Bicgstab).*
- *Two time-consuming kernels:*
  - $q = M^*p$
  - $r(p) = g - M^*p$

# Observations

- $r(p) = g - M^*p$   
 $= \hat{x}_2(p) - x_2(p)$
- $r(0) = g$   
 $= \hat{x}_2(0) - x_2(0)$
- $q = M^*p$   
 $= r(0) - r(p)$

# PSPIKE

## Stage 1 –

- Extraction of an effective “banded” preconditioner  $M$  – with or without reordering.
- $\text{norm}(M, 'fro') = (1 - \varepsilon) \text{norm}(A, 'fro')$ ;
- “bandwidth”  $\leq \beta(n)$ .

## Stage 2 –

- Use an outer Krylov subspace method (*BiCGstab*) to solve  $Ax = f$  with preconditioner  $M$ .
- *A modified Pardiso* (by O. Schenk) is a major kernel for solving  $Mz = r$ .
- Classes of banded preconditioners:  
narrow-banded, generalized-banded, wide-banded



# *Computing Platform*

- *Intel Cluster*

- *Westmere X5670, 2.93 GHz*

- *Infiniband interconnection*

- *12-core nodes*

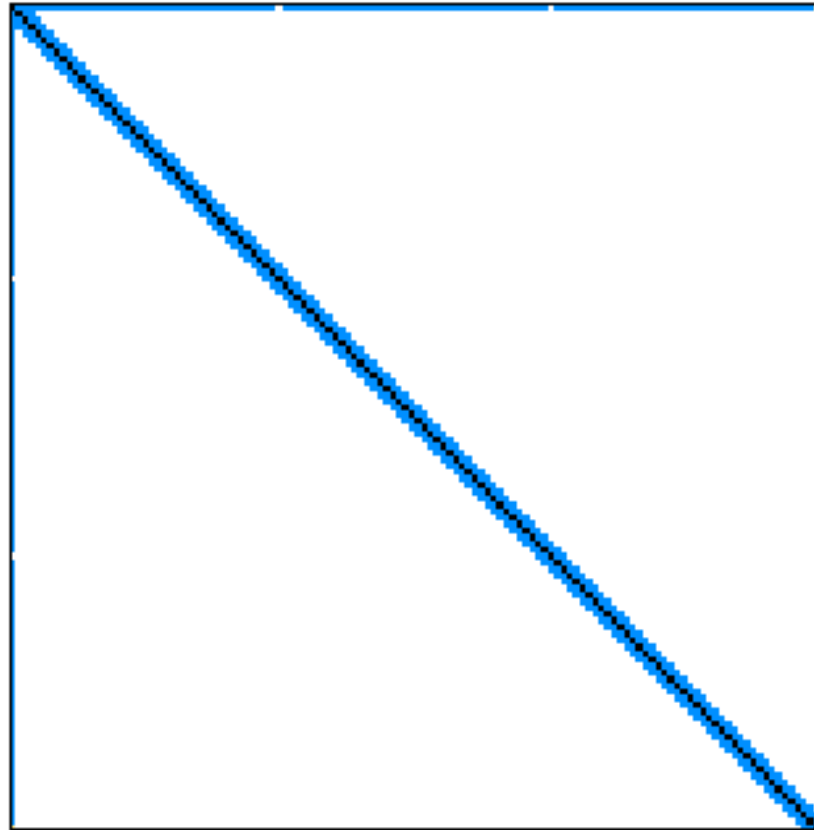
*a. PSPIKE-NBP*

# *UFL – Rajat31 (circuit simulation)*

$N \sim 4.7 M$

$nnz \sim 20 M$

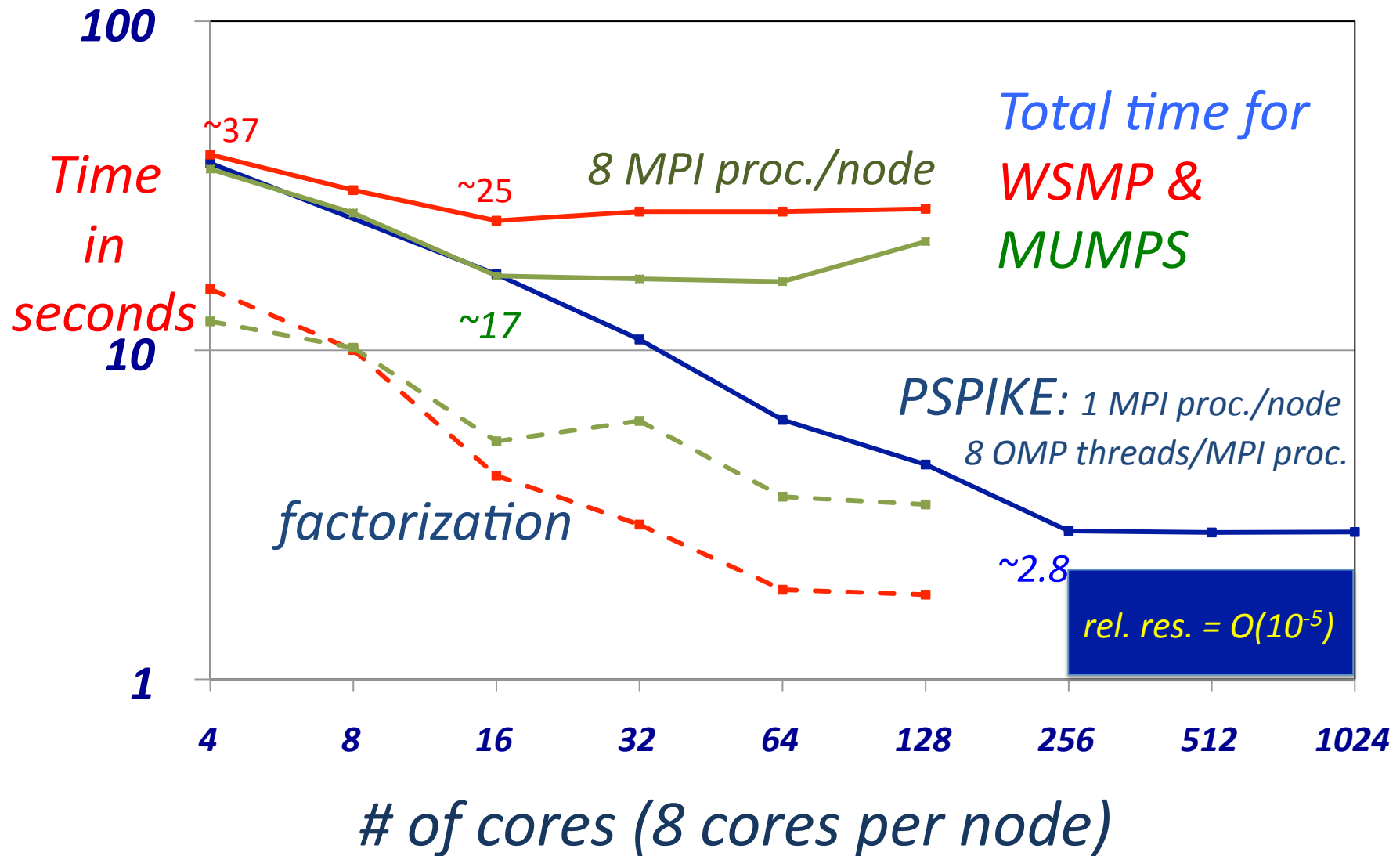
*nonsymmetric*



# *Scalability of PSPIKE vs. direct solvers (for Rajat31)*

- *Direct solvers:*
  - *WSMP*
    - *matrix input centralized on Node 0*
  - *MUMPS*
    - *matrix input distributed by block rows*
    - *solution vector distributed*
    - *ParMETIS*
- *PSPIKE: (bandwidth of preconditioner = 5)*
  - *TraceMIN-Fiedler*
  - *matrix input distributed by block rows*

# PSPIKE – WSMP – MUMPS



# MEMS simulation benchmark 1

System size:

$N = 11,333,520$

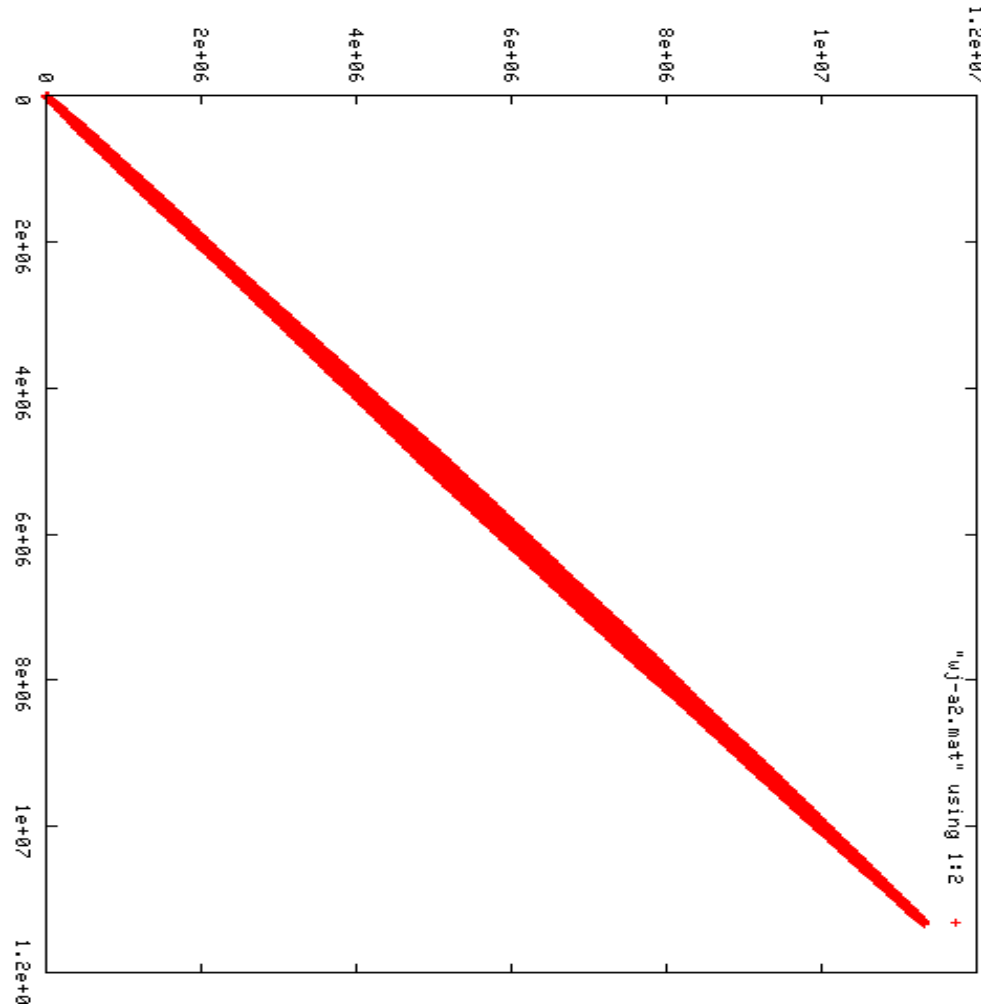
# of nonzeros:

61,026,416

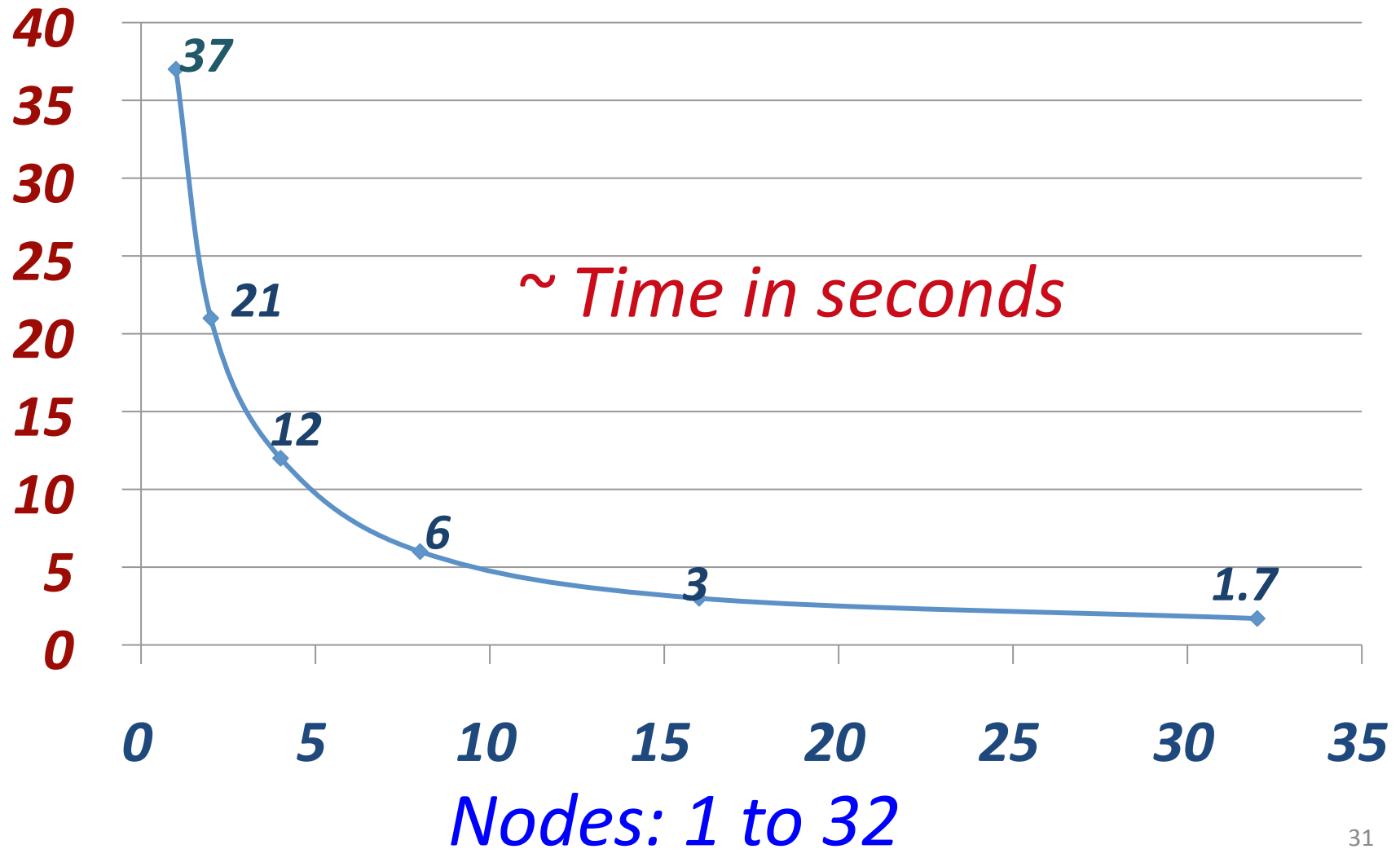
bandwidth:

334,613

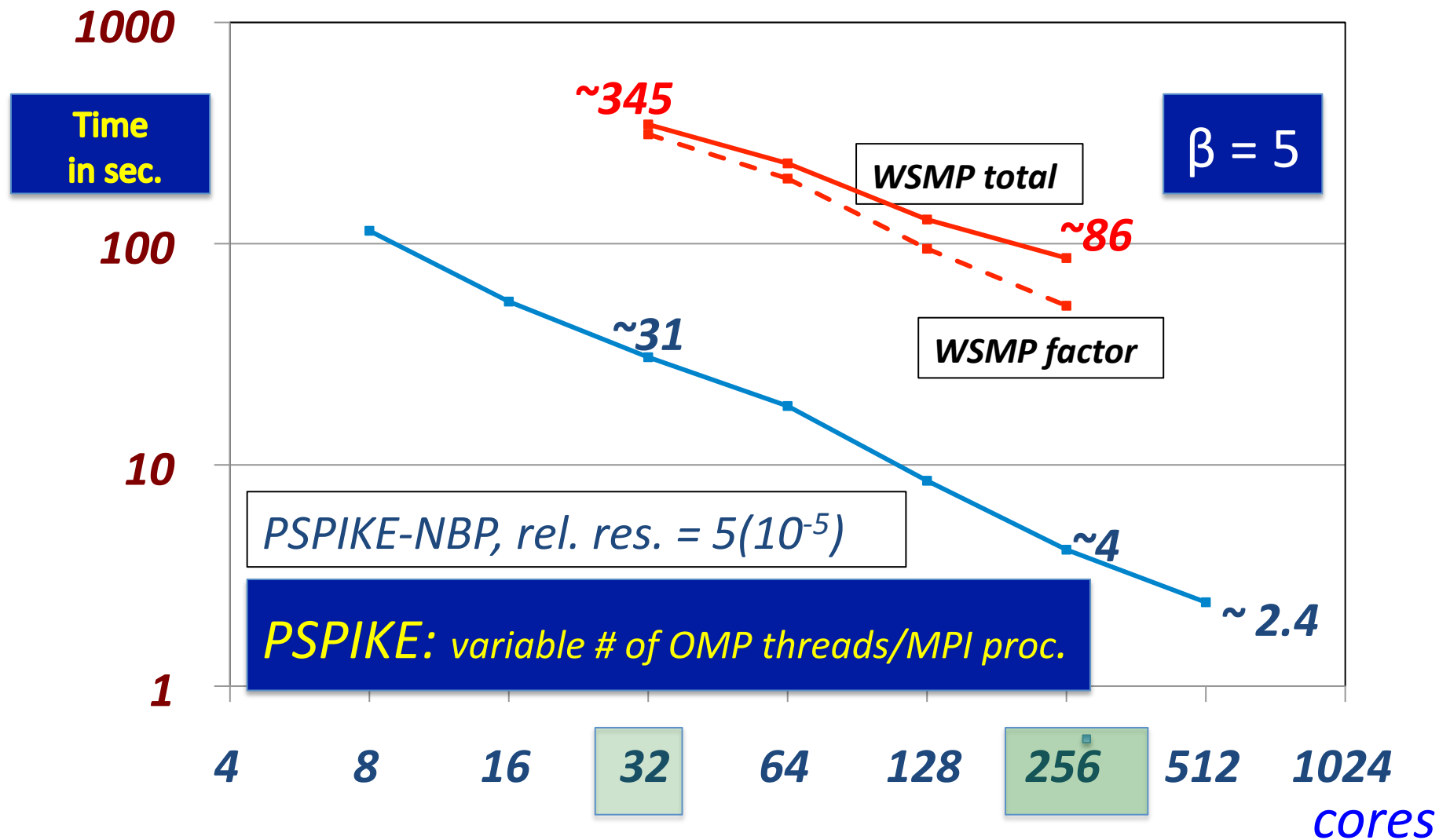
stopping criterion:  
 $\text{rel. res.} = O(10^{-2})$



# Scalability of TraceMin-Fiedler




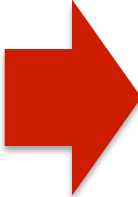
# MEMS Simulation – benchmark-1





# *Scalability of PSPIKE vs. Trilinos*

## *Intel Harpertown*

- *Strong scalability of PSPIKE*  
*Fixed problem size – 1 to 64 nodes (or 8 to 512 cores)*
- *Comparison with AMG-preconditioned Krylov subspace solvers in:*
  - *Hypre (LLNL)*
  - *Trilinos-ML (Sandia)* 
    - *Smoother –*
      - *Chebyshev*  *fastest*
      - *Jacobi*
      - *Gauss-Seidel*

# Speed Improvement over Trilinos-ML

1 2 4 8 16 32 64(nodes)

$\text{Time (Trilinos-ML)} \div \text{Time (PSPIKE)}$

PSPIKE:  $k$  threads per MPI process

break-even @ 4 nodes

100

10

1

0.1

Intel Harpertown

# of nodes  $k$

1 to 4 1

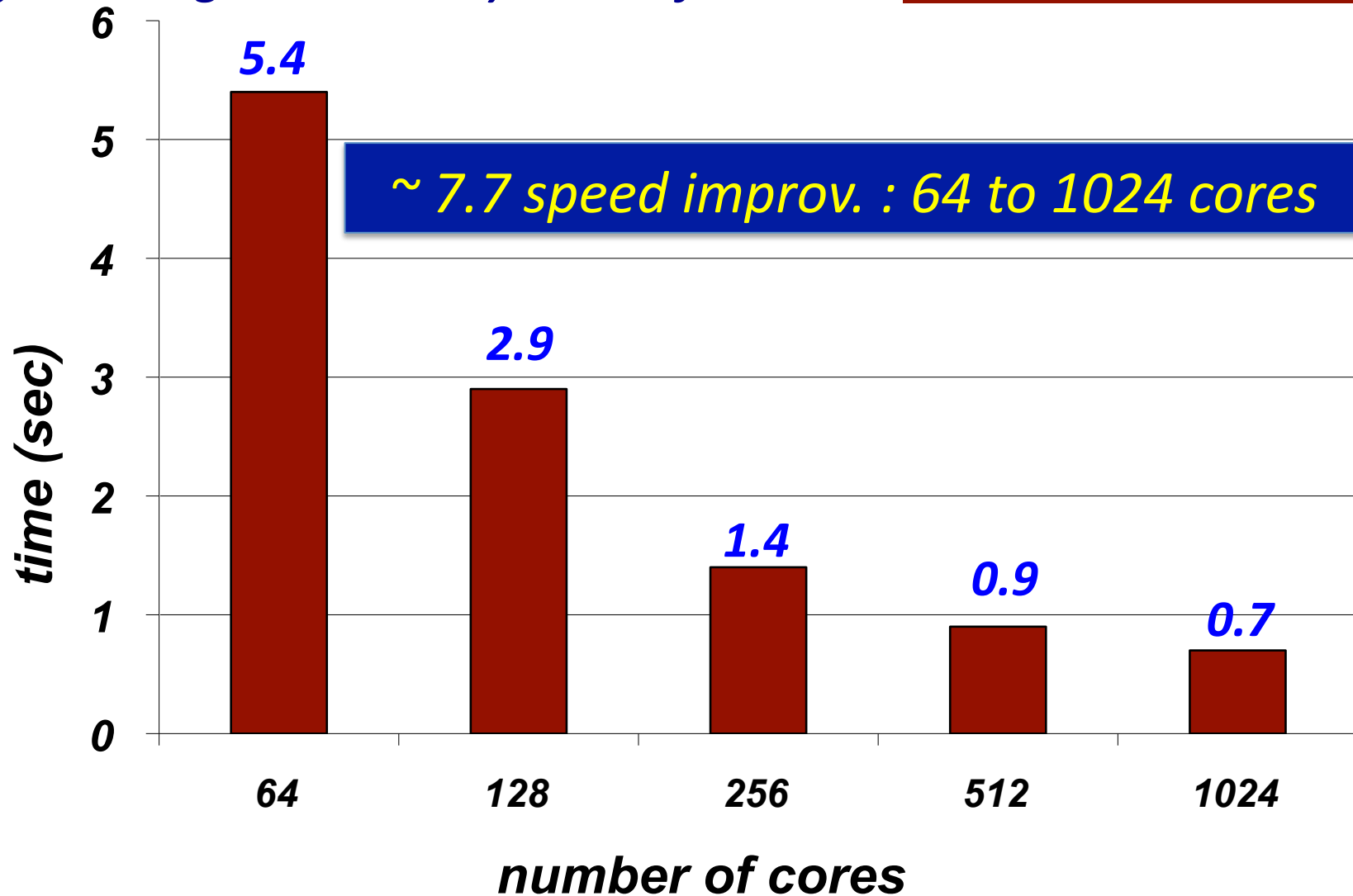
8 to 16 4

> 16 8

MEMS benchmark 1

# *Strong Scalability on Intel Nehalem*

*for a larger MEMS system of order  $\sim$  23M (benchmark 2)*



*b. PSPIKE-WBP*

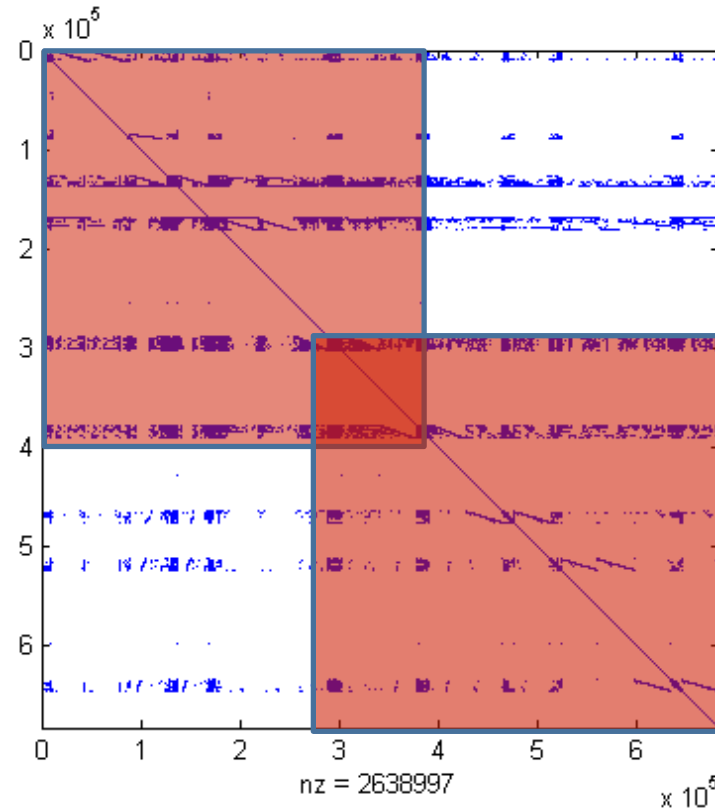
# PSPIKE for *ASIC680K* system

relative residual

*Pardiso*:  $O(10^{-9})$

*ILU+BiCGstab*:  $O(10^{-7})$

*PSPIKE-WBP*:  $O(10^{-7})$

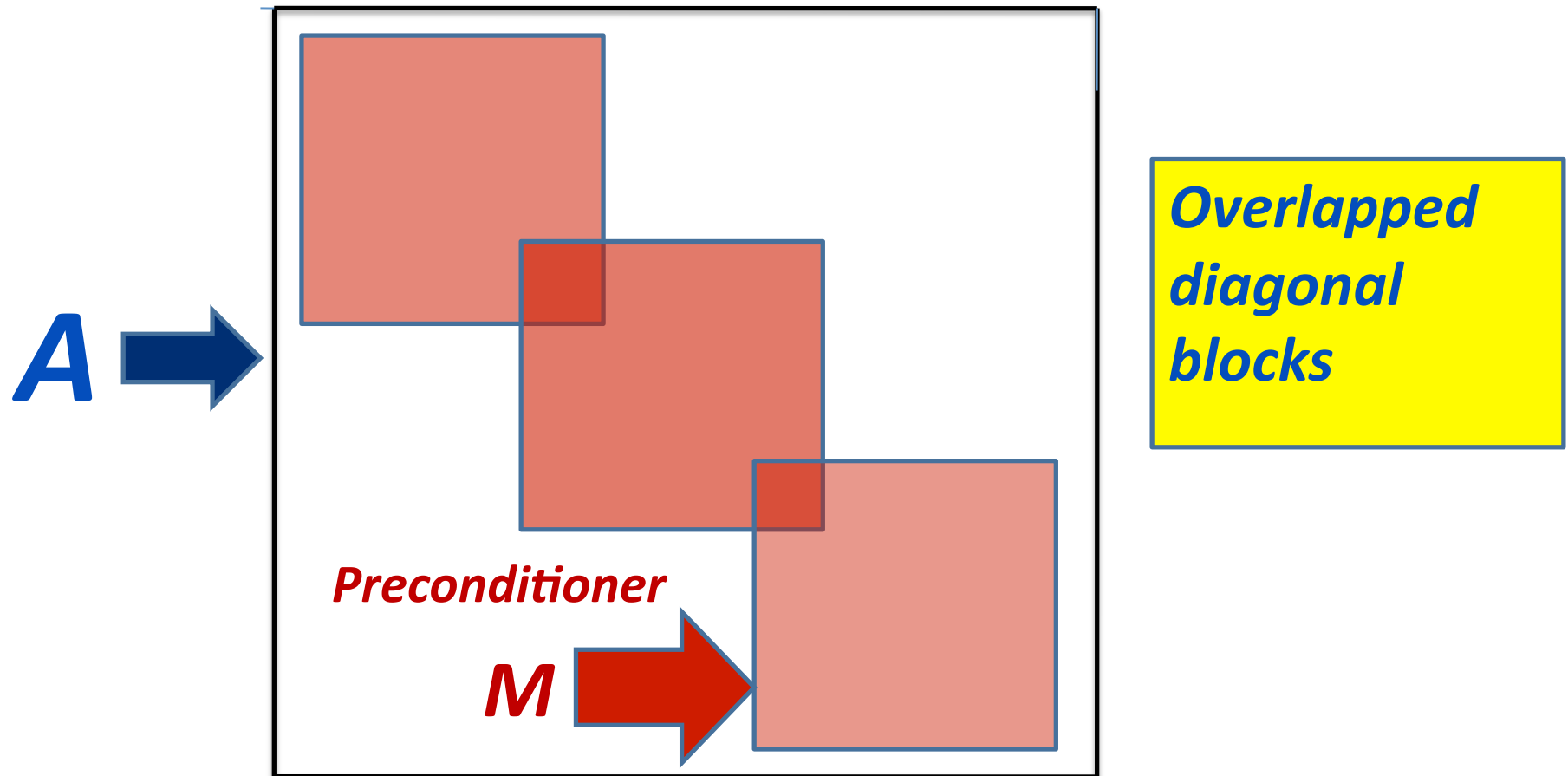


**Preconditioner  
consisting of two  
overlapped blocks  
(overlap = 11)**

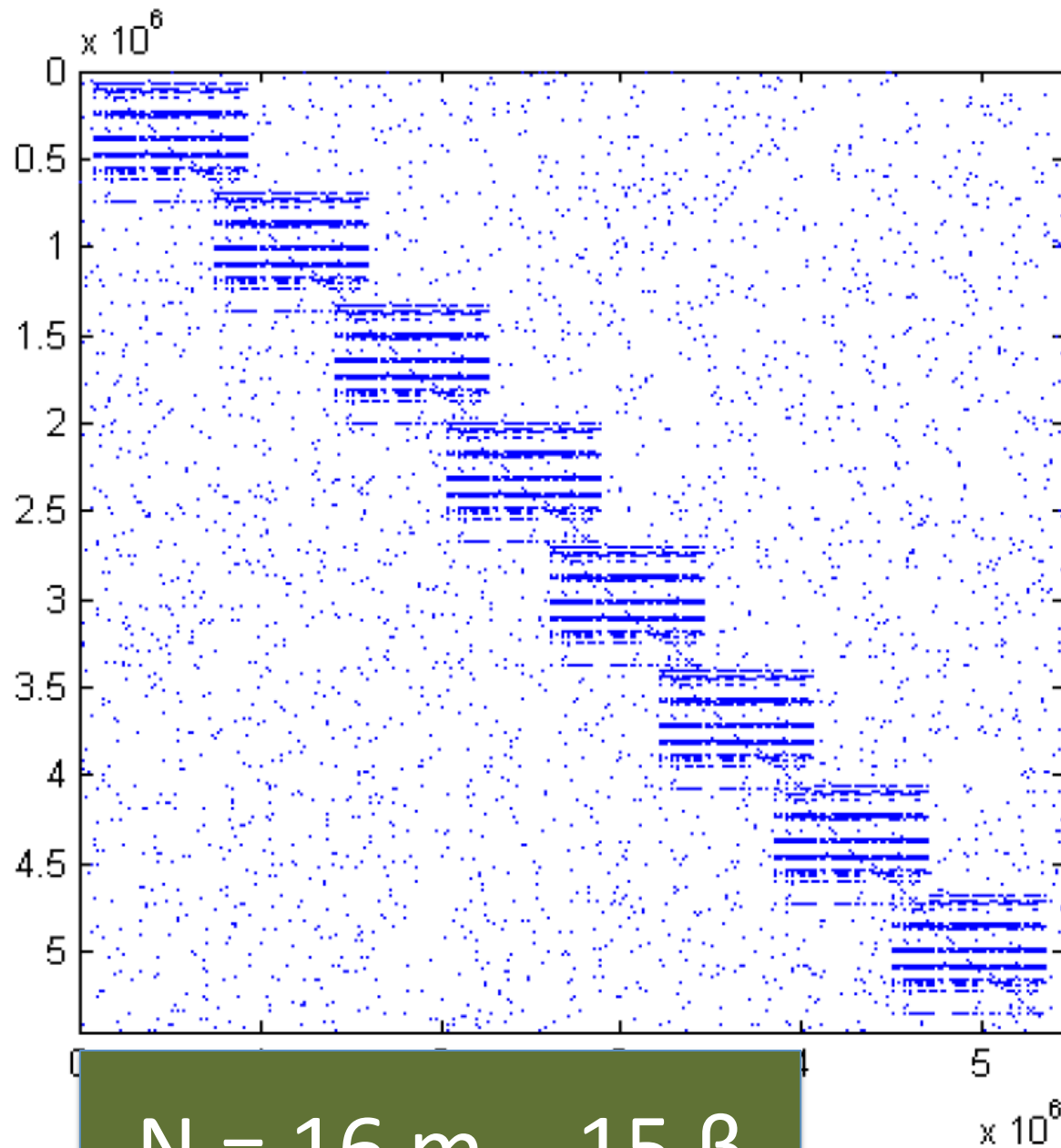
Time on 1 node: *Pardiso* – 23 (sec).

*ILU-BiCGstab* – 34 (sec).

Time on 2 nodes: *PSPIKE* – 6 (sec).



*PSPIKE-WBP*



***PSPIKE-WBP***

*ASIC\_680k\_8  
nonsymmetric  
size:  $\sim 5.5 M$   
nnz:  $\sim 31.0 M$*

*Preconditioner:  
16 overlapped  
diagonal blocks  
overlap size = 100*

# *PSPIKE-WBP vs. ILU-preconditioned BiCGstab*

- *ILUT-BiCGstab (one core):*
  - *drop tolerance =  $10^{-3}$*
  - *fill\_in per row = 10%*
  - *16 iterations, rel. res. =  $O(10^{-6})$*
  - *Time  $\sim 276$  sec.*
- *PSPIKE-WBP :*
  - *1 iteration, rel. res =  $O(10^{-9})$*



# *PSPIKE-WBP (multiple nodes)*

vs.

*ILU-preconditioned BiCGstab (one core)*

<i># of cores</i>	<i>32 (4 nodes)</i>	<i>64 (8 nodes)</i>	<i>128 (16 nodes)</i>
<i>Time (sec.)</i>	<i>113</i>	<i>31</i>	<i>9</i>
<i>Speed improv. over ILU-BiCGstab (one core)</i>	<i>2.4</i>	<i>8.9</i>	<i>30.7</i>

*Thank You!*