Nested Dissection Approach to Sparse Matrix Partitioning

SIAM PP08 — 3/12/2008

Erik Boman — Sandia National Laboratories
Michael Wolf — University of Illinois at Urbana-Champaign

Parallel Sparse Matrix-Vector Multiplication

$$
\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \end{bmatrix}
=
\begin{bmatrix}
1 & 6 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 1 & 9 & 0 & 5 & 0 & 0 & 0 \\
0 & 8 & 1 & 7 & 0 & 0 & 0 & 0 \\
0 & 0 & 2 & 1 & 0 & 0 & 0 & 7 \\
0 & 0 & 0 & 0 & 1 & 8 & 0 & 0 \\
4 & 0 & 0 & 0 & 3 & 1 & 3 & 0 \\
0 & 0 & 0 & 6 & 0 & 9 & 1 & 4 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & 1
\end{bmatrix}
\begin{bmatrix} 1 \\ 2 \\ 4 \\ 3 \\ 1 \\ 4 \\ 2 \\ 1 \end{bmatrix}
$$

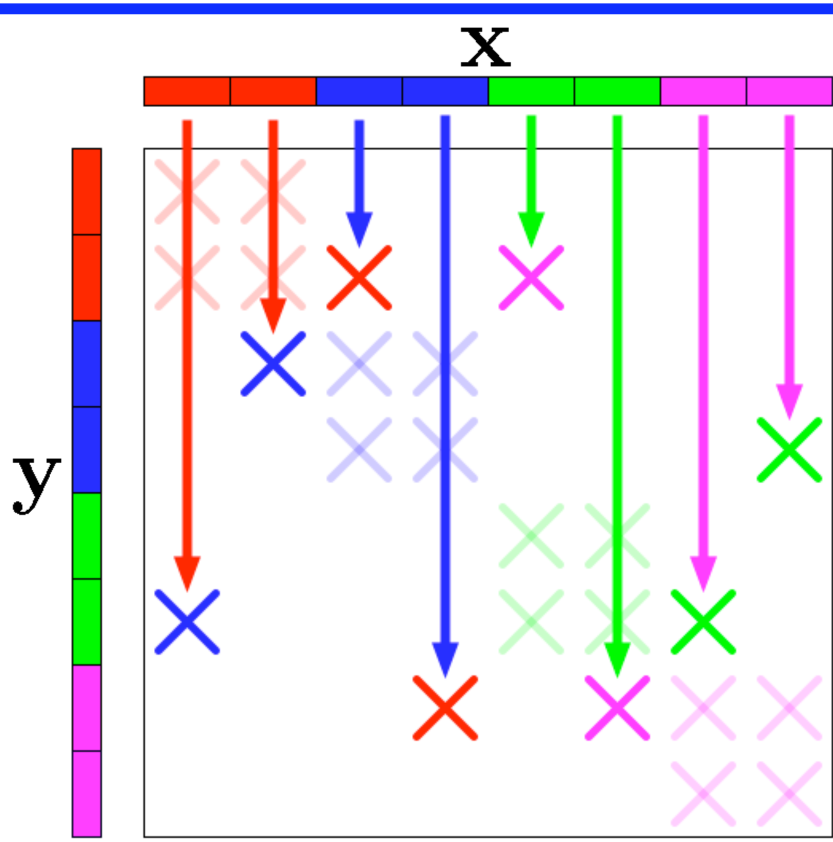
$$\mathbf{y} = \mathbf{A}\mathbf{x}$$

- Partition matrix nonzeros
- Partition vectors

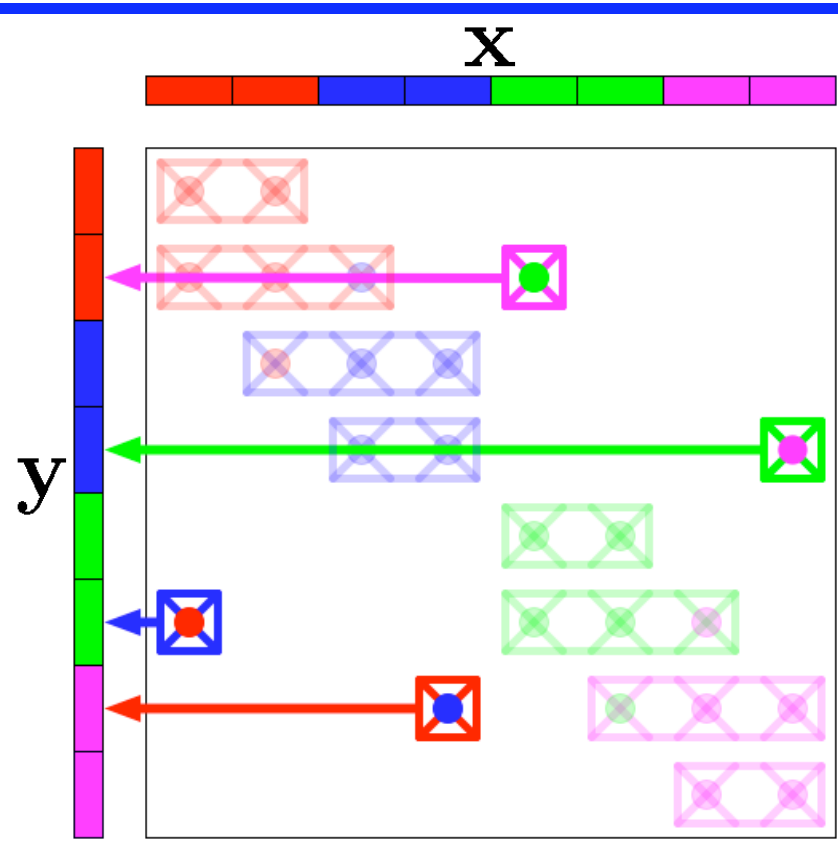
Objective

- Ideally we minimize total run-time
- Settle for easier objective
  - Work balanced
  - Minimize total communication volume
- Can partition matrices in different ways
  - 1-D
  - 2-D
- Can model problem in different ways
  - Graph
  - Bipartite graph
  - Hypergraph

Parallel MatVec Multiplication Communication



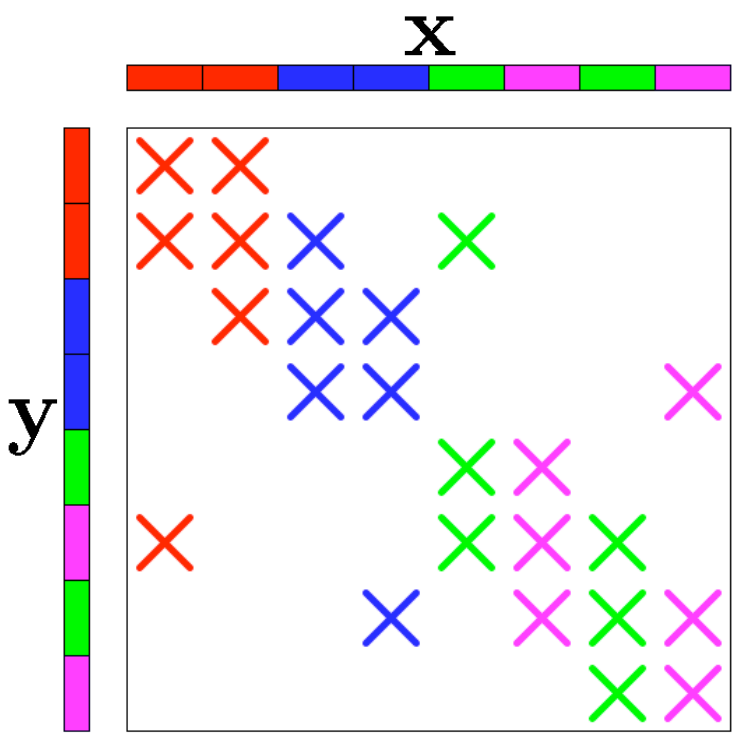
"fan-out"



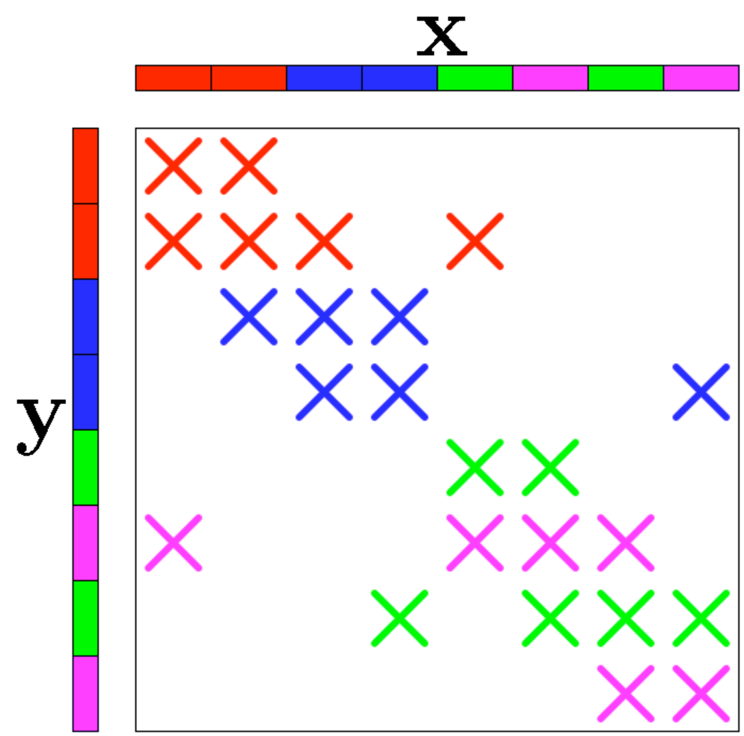
"fan-in"

- x_j sent to remote processes that have nonzeros in column j
- Partial inner-products sent to process that owns vector element y_i

1-D Partitioning



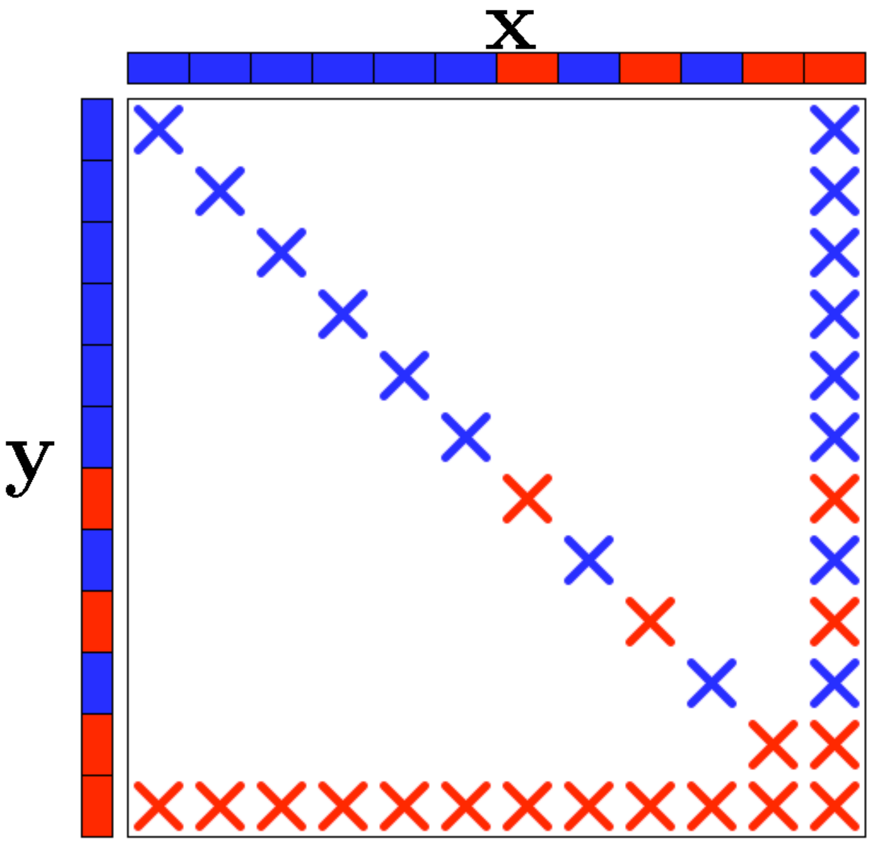
1-D Column



1-D Row

- Each process assigned nonzeros for set of columns
- Each process assigned nonzeros for set of rows

When 1-D Partitioning is Inadequate



“Arrowhead” matrix

n=12

nnz=34 (18,16)

volume = 9

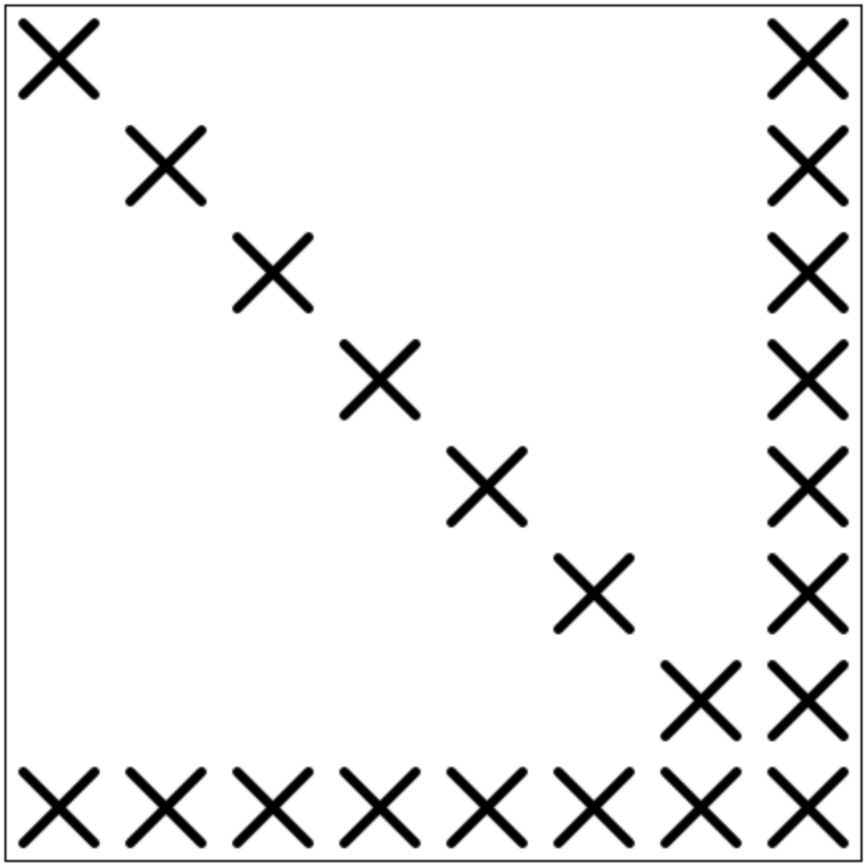
- For any 1-D bisection of nxn arrowhead matrix:
  - nnz = 3n-2
  - Volume ≈ (3/4)n
- O(p) volume partitioning possible

2-D Partitioning Methods

- More flexibility in partitioning
- No particular partition for given row or column
- More general sets of nonzeros assigned partitions

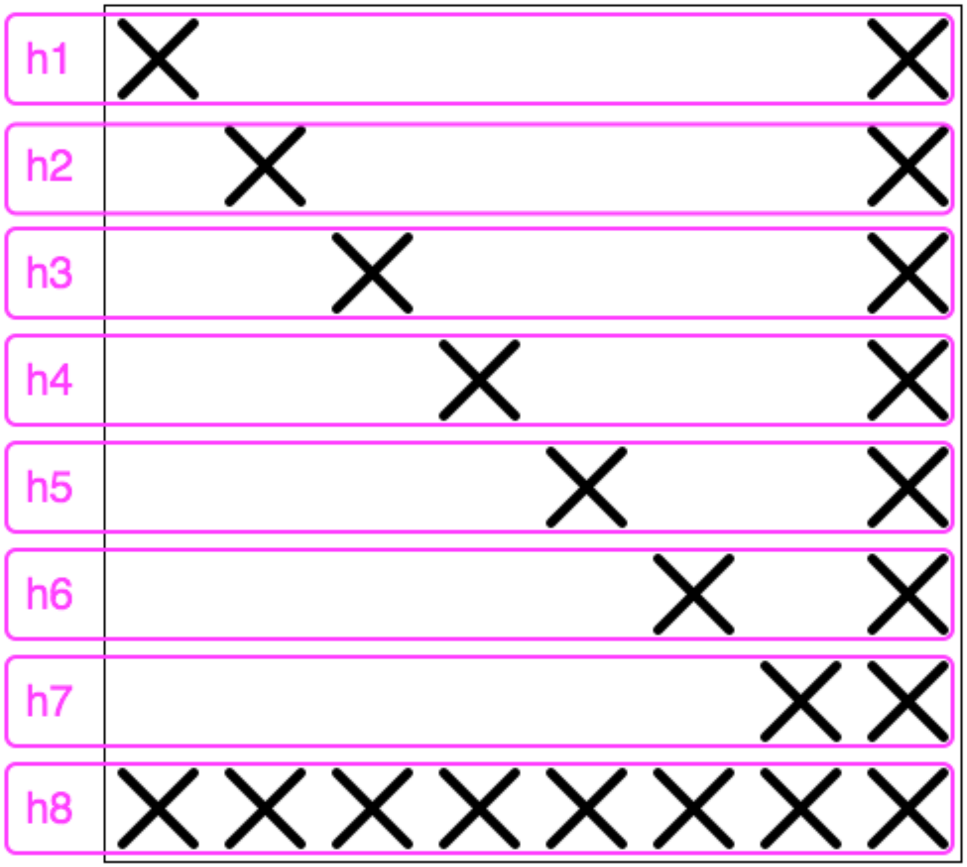
- Fine-grain hypergraph model
  - Ultimate flexibility
  - Assign each nz separately
- Graph model for symmetric 2-D partitioning
- Vertex separator partitioning method

Fine-Grain Hypergraph Model



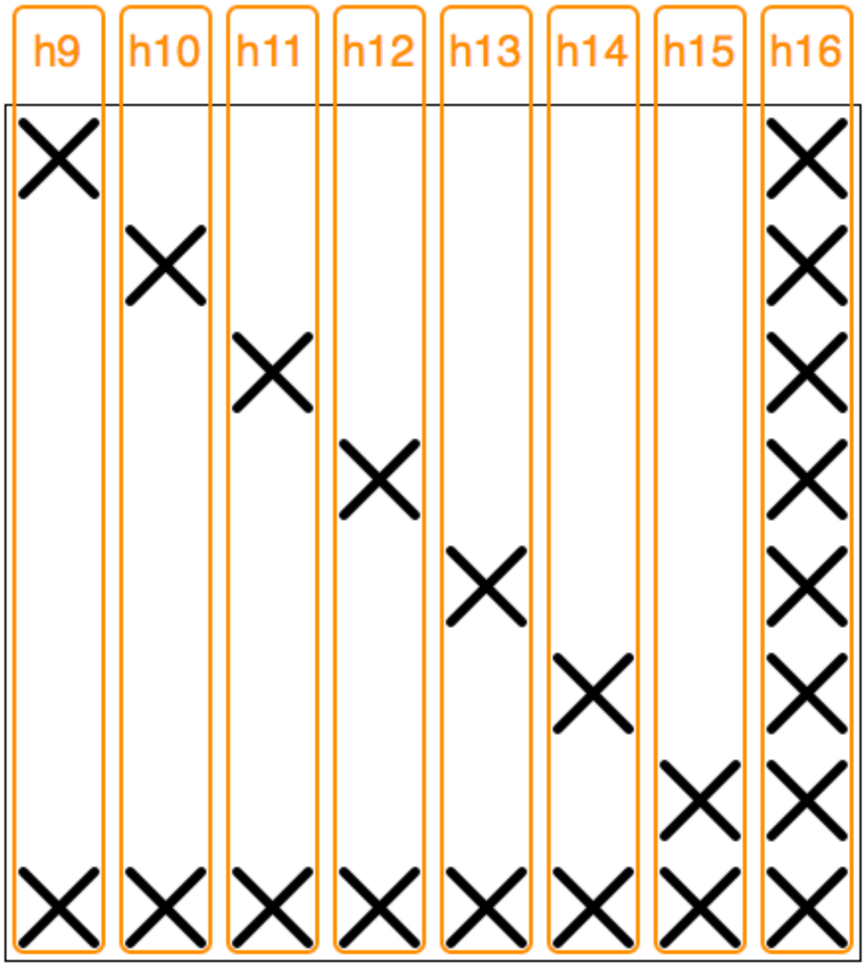
- Catalyurek and Aykanat (2001)
- Nonzeros represented by vertices in hypergraph

Fine-Grain Hypergraph Model



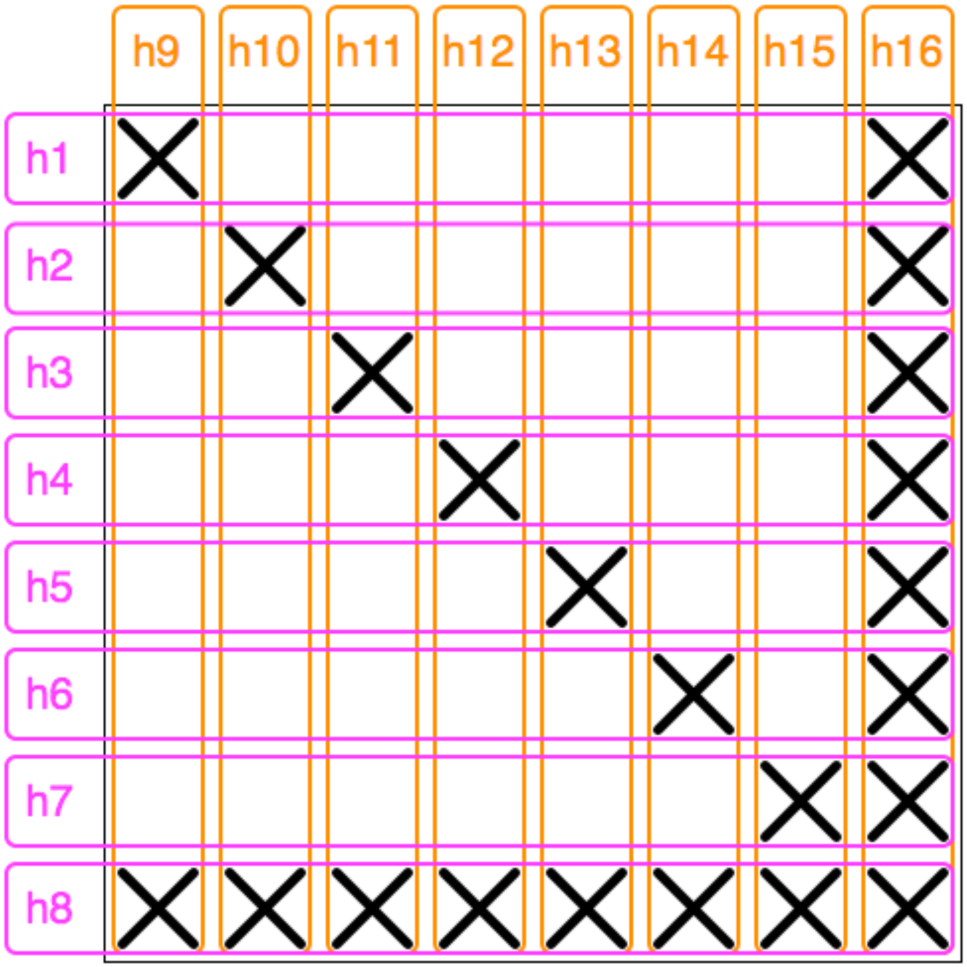
- Rows represented by hyperedges

Fine-Grain Hypergraph Model



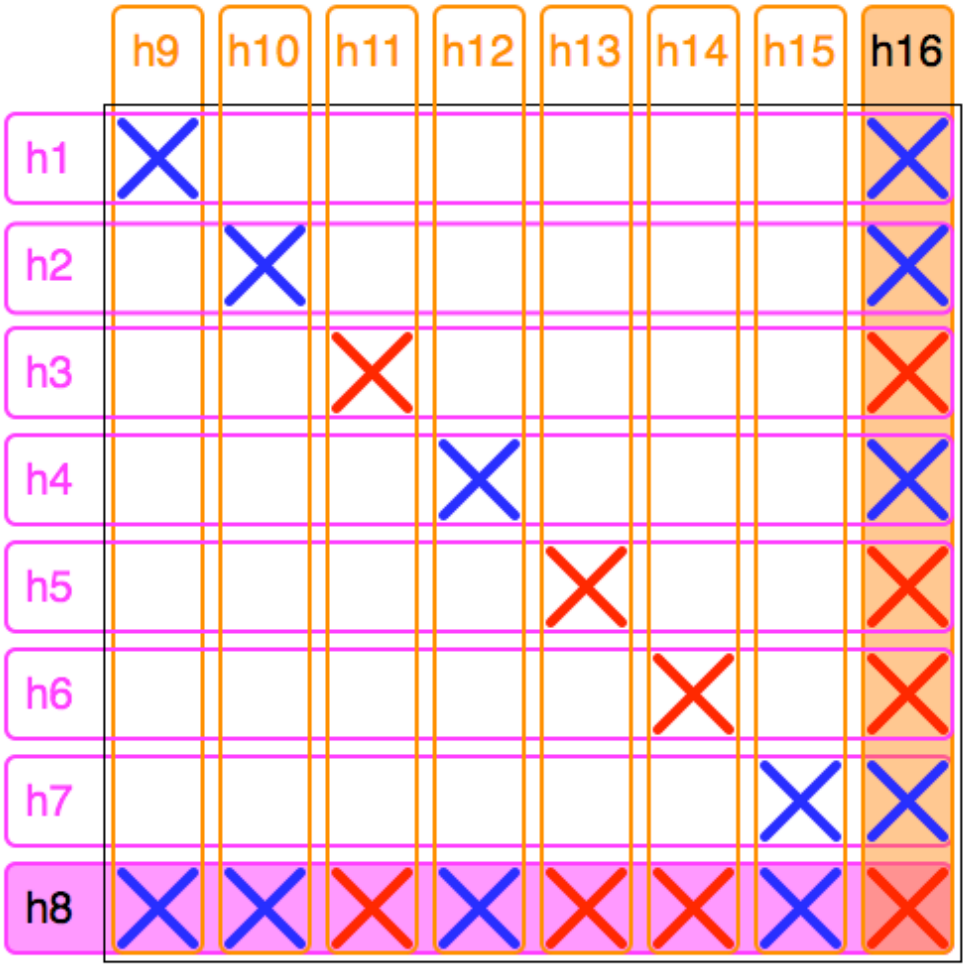
- Columns represented by hyperedges

Fine-Grain Hypergraph Model



- 2n hyperedges

Fine-Grain Hypergraph Model



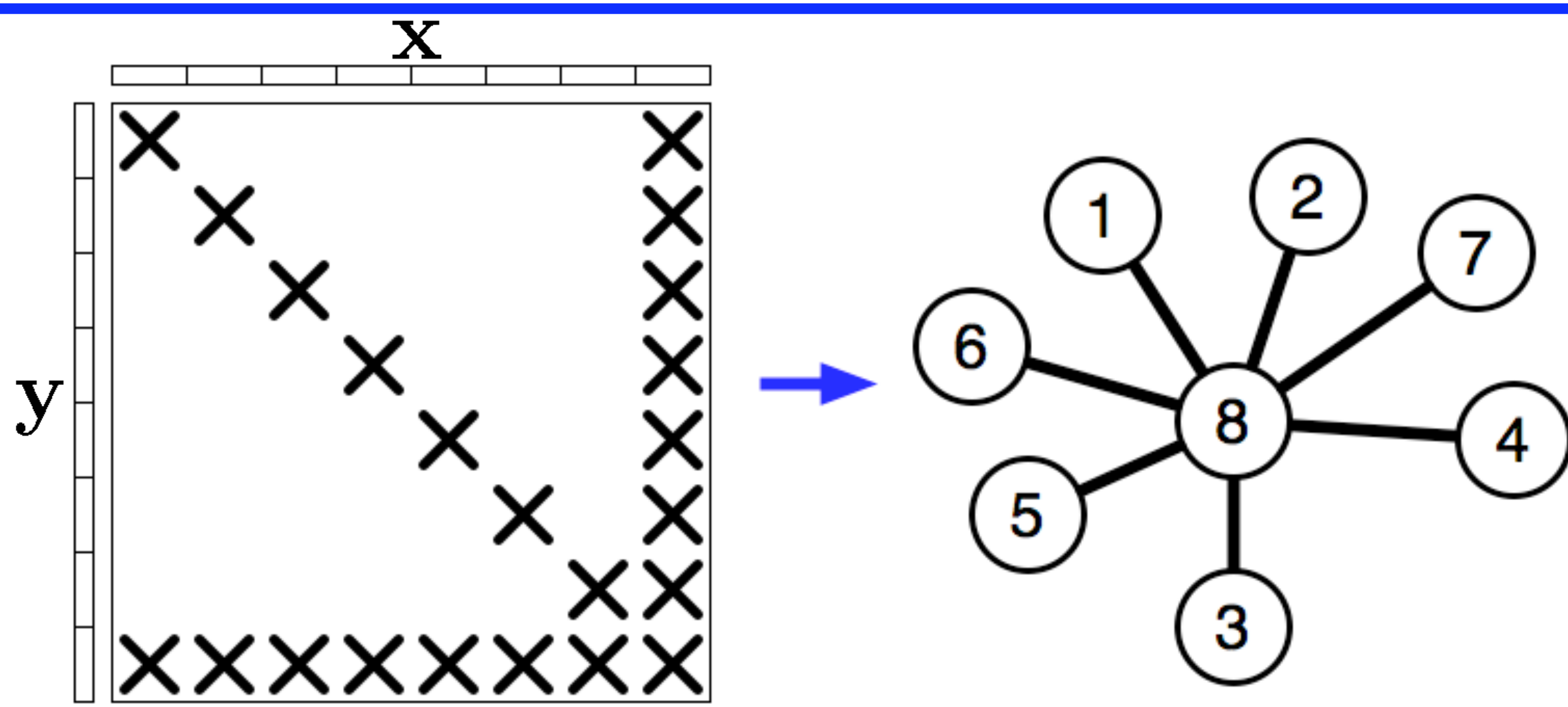
k=2, volume = cut = 2

- Partition vertices into k equal sets
- For k=2
  - Volume = number of hyperedges cut
- Minimum volume partition when optimally solved
- Larger NP-hard problem than 1-D

Graph Model for Symmetric 2-D Partitioning

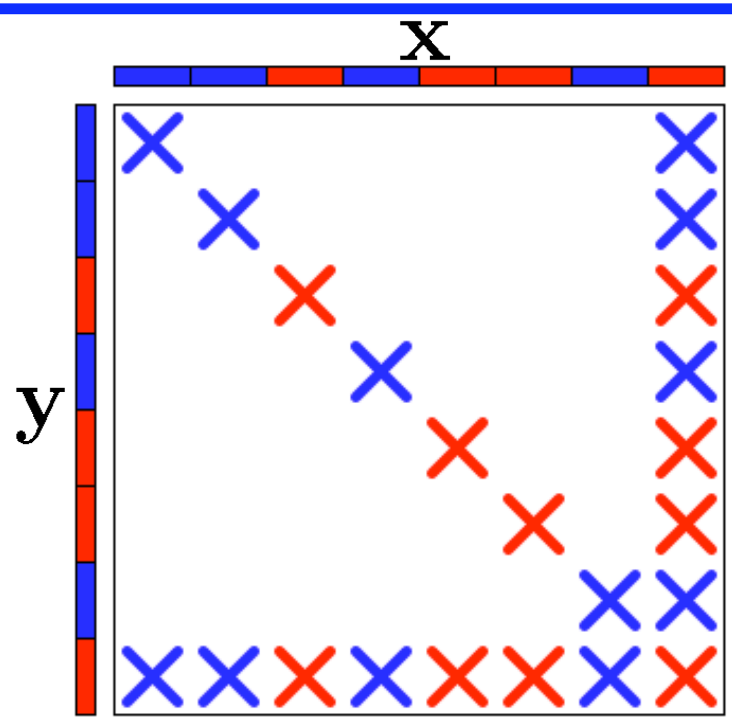
- Given symmetric matrix A
- Symmetric partition
  - $a(i,j)$ and $a(j,i)$ assigned same partition
  - Input and output vectors have same distribution
- Corresponding graph $G(V,E)$
  - Vertices correspond to vector elements
  - Edges correspond to off-diagonal nonzeros

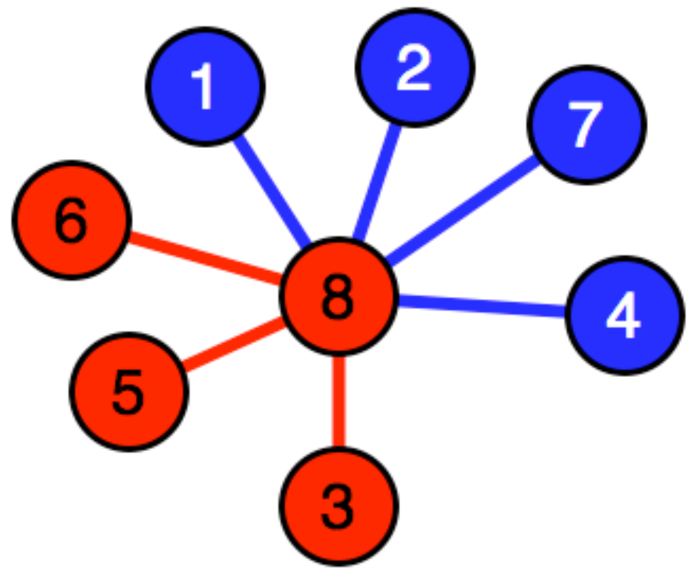
Graph Model for Symmetric 2-D Partitioning



- Corresponding graph G(V,E)
  - Vertices correspond to vector elements
  - Edges correspond to off-diagonal nonzeros

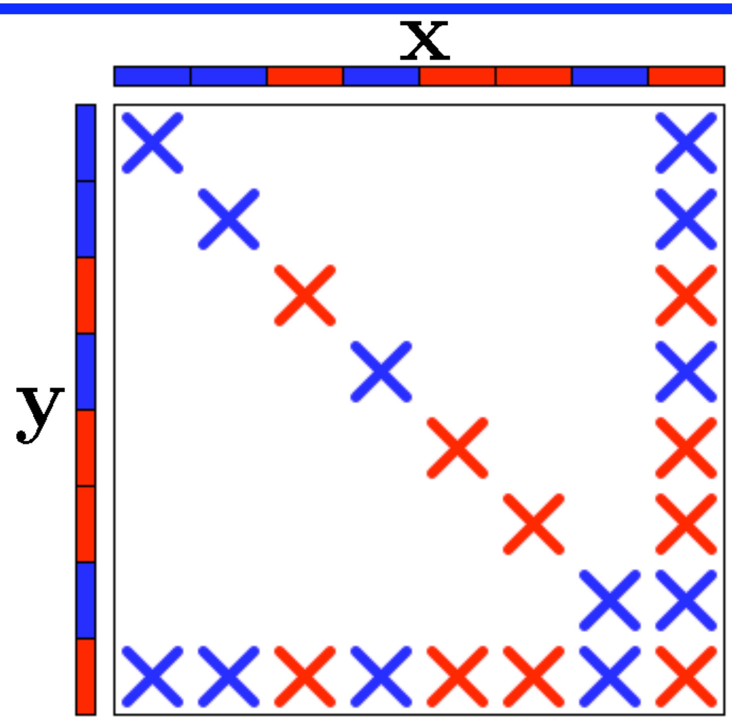
Graph Model for Symmetric 2-D Partitioning

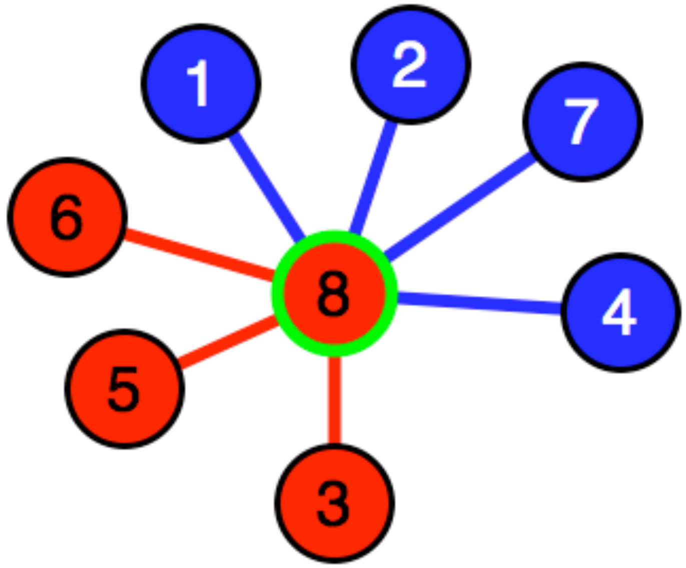




- Symmetric 2-D partitioning
  - Partition both V and E
  - Gives partition of both matrix and vectors

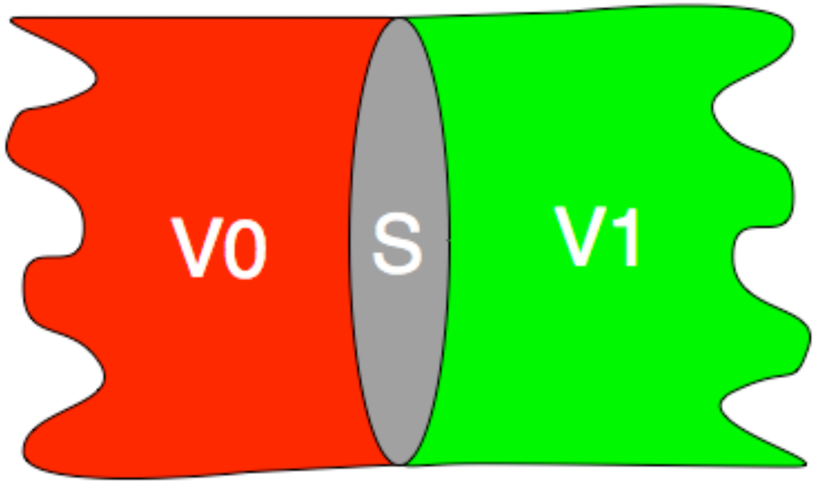
Communication in Graph Model





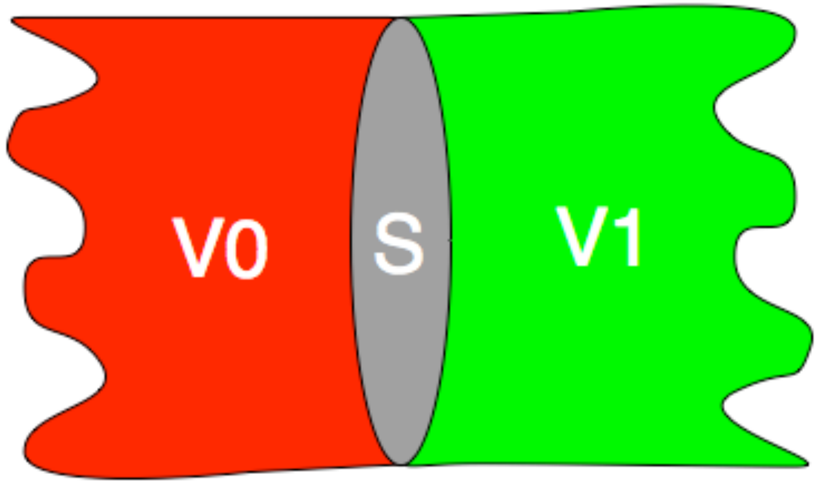
- Communication is assigned to vertices
- Vertex incurs communication iff incident edge is in different partition
- Want small vertex separator -- $S=\{V_8\}$

Nested Dissection Partitioning Method - Bisection



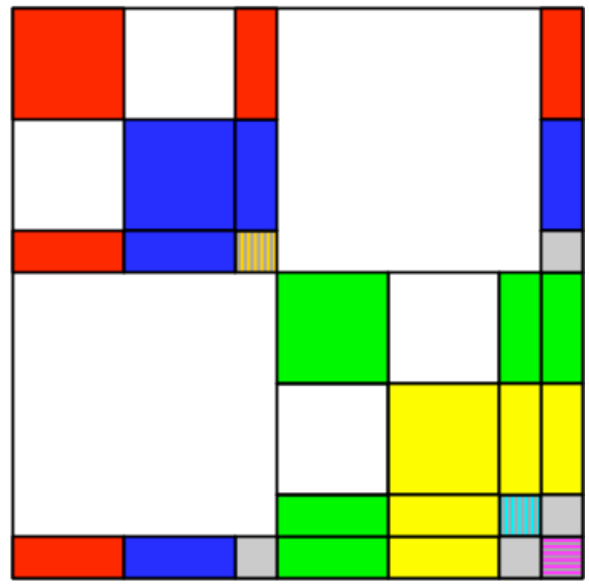
- Suppose A is symmetric
- Let $G(V,E)$ be graph of A
- Find small, balanced separator S
  - Yields vertex partition $V = (V0, V1, S)$
- Partition the edges
  - E0 = {edges that touch a vertex in V0}
  - E1 = {edges that touch a vertex in V1}

Nested Dissection Partitioning Method - Bisection

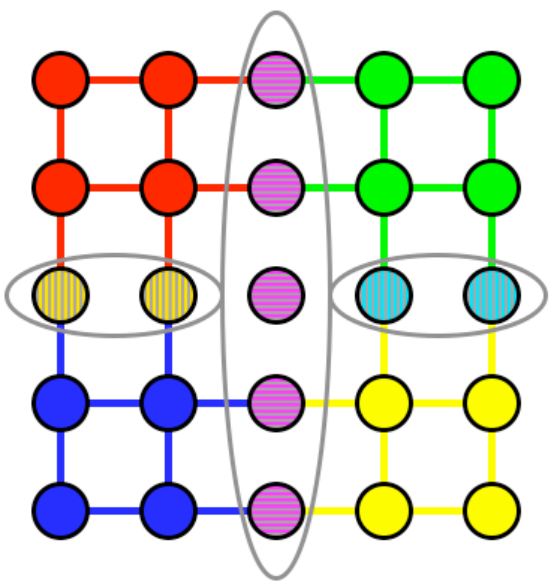


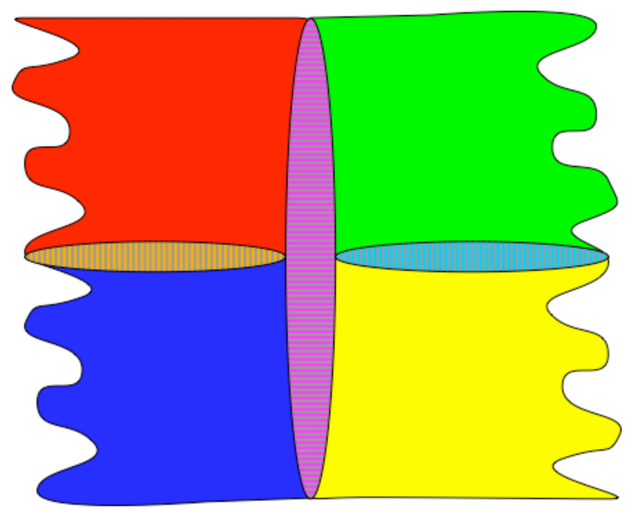
- Vertices in S and corresponding edges
  - Can be assigned to either partition
  - Can use flexibility to maintain balance
- Communication Volume = 2*|S|
  - Regardless of S partitioning
  - |S| in each phase

Nested Dissection Partitioning Method



- Recursive bisection to partition into >2 partitions
- Use nested dissection!

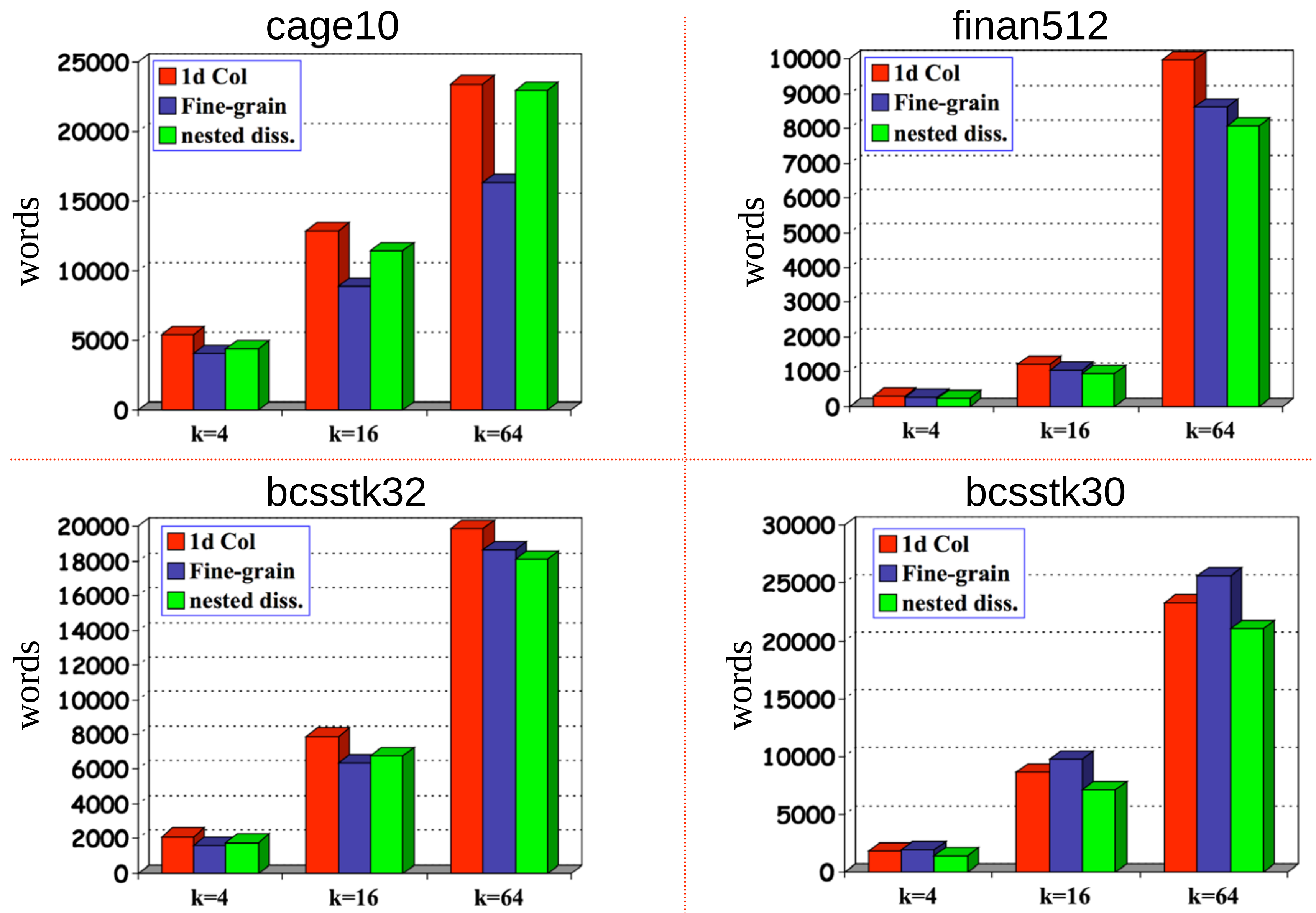




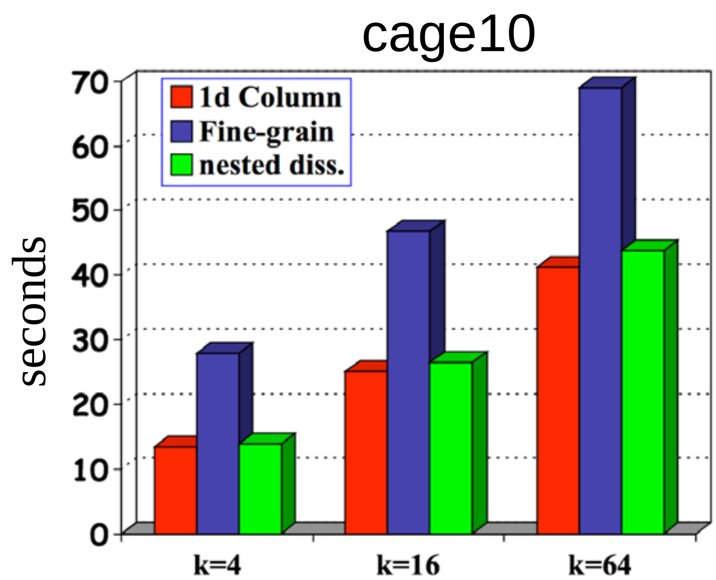
Numerical Experiments

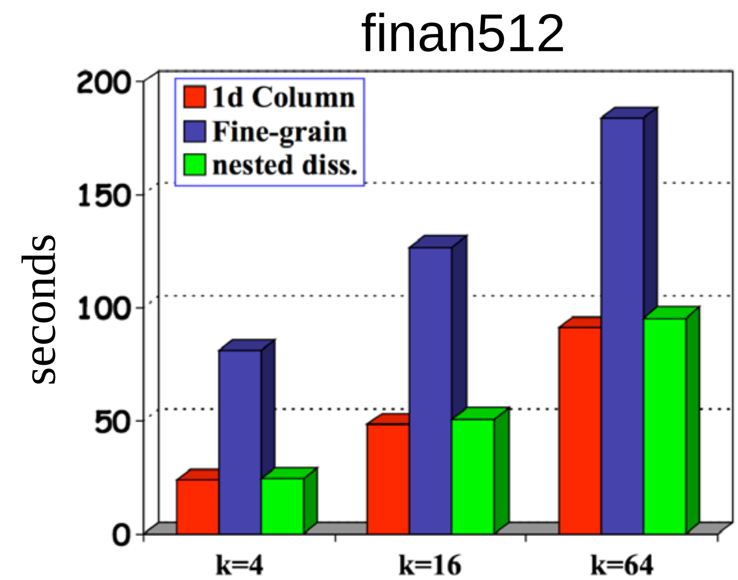
- Compared 3 methods
  - 1-D hypergraph partitioning
  - Fine-grain hypergraph partitioning
  - Nested dissection partitioning
- PaToH for hypergraph partitioning
- Finding separators
  - 1-D hypergraph partitioning
  - ODU minimum vertex cover software
    - Florin Dobrian, Mahantesh Halappanvar, and Alex Pothen
- Symmetric and nonsymmetric matrices
  - Mostly from Prof. Rob Bisseling (Utrecht Univ.)
- k = 4, 16, 64 partitions

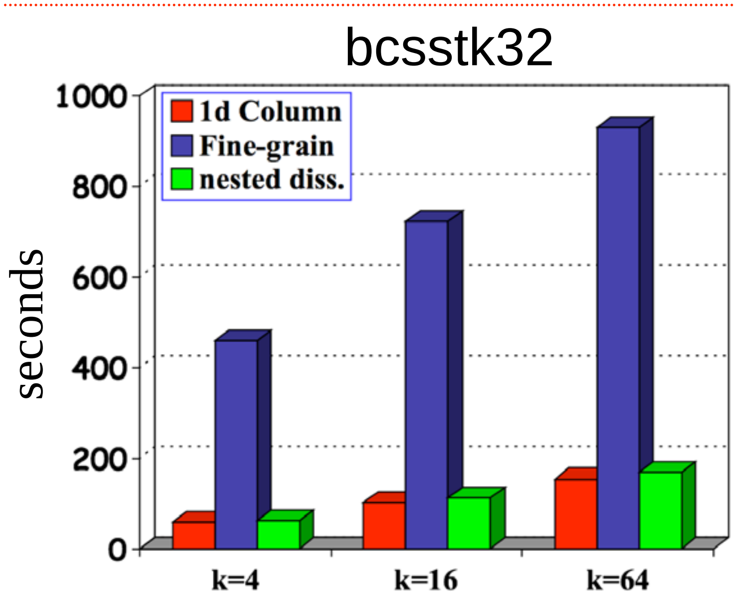
Communication Volume - Symmetric Matrices

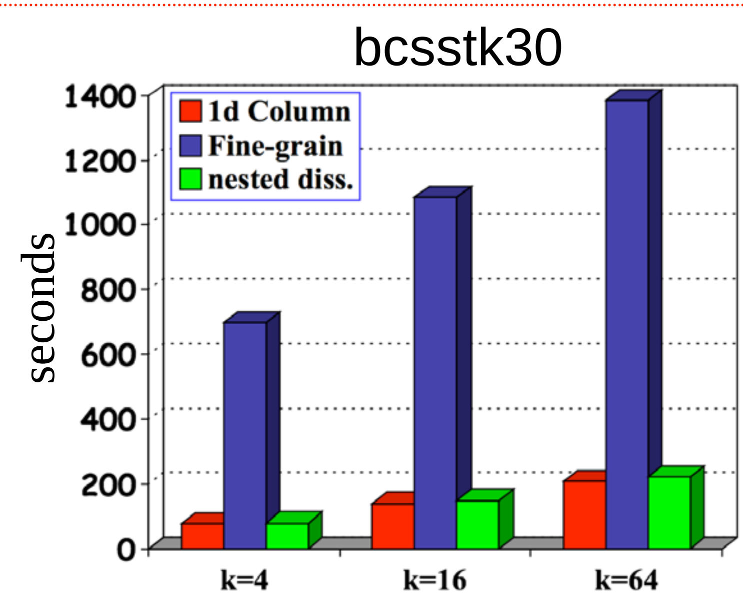


Estimated Runtimes



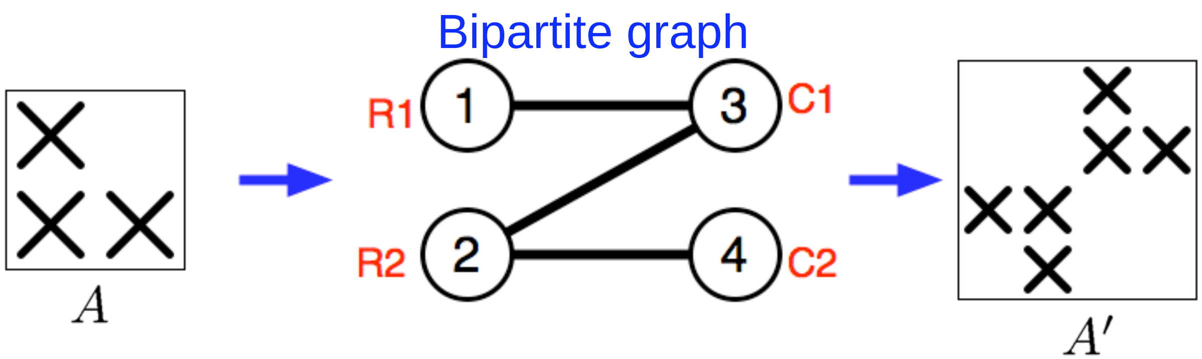






Nonsymmetric Matrices

- Given nonsymmetric matrix A
- Construct bipartite graph $G'(R,C,E)$
  - R vertices correspond to rows, C vertices to columns
  - E correspond to nonzeros
  - Can be represented by symmetric adjacency matrix

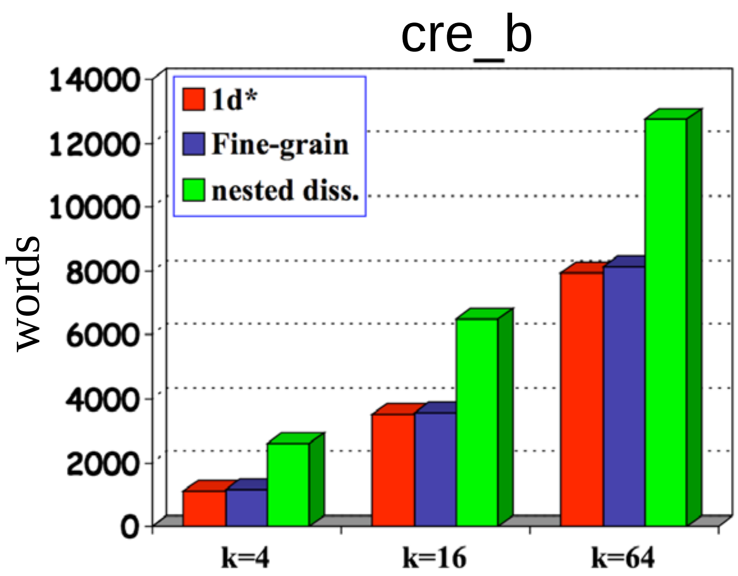


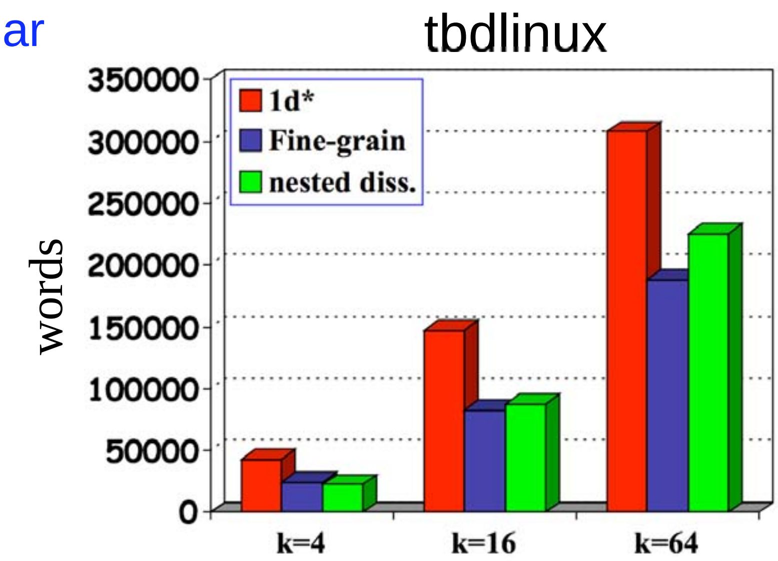
$$A' = \begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix}$$

- Apply nested dissection approach to G'
  - Use same algorithm as for symmetric case

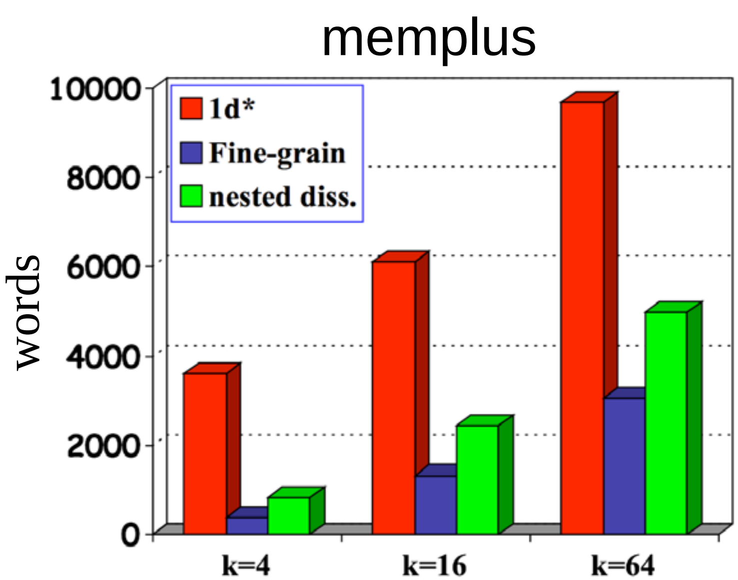
Communication Volume - Nonsymmetric Matrices

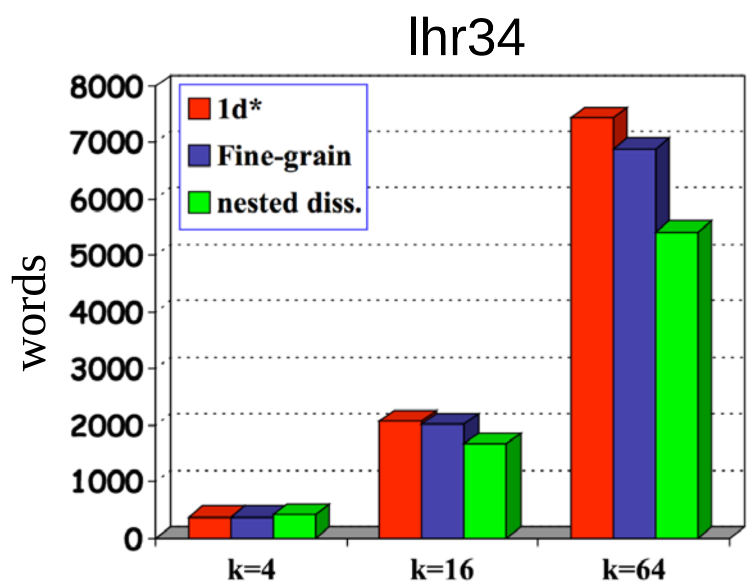
Rectangular



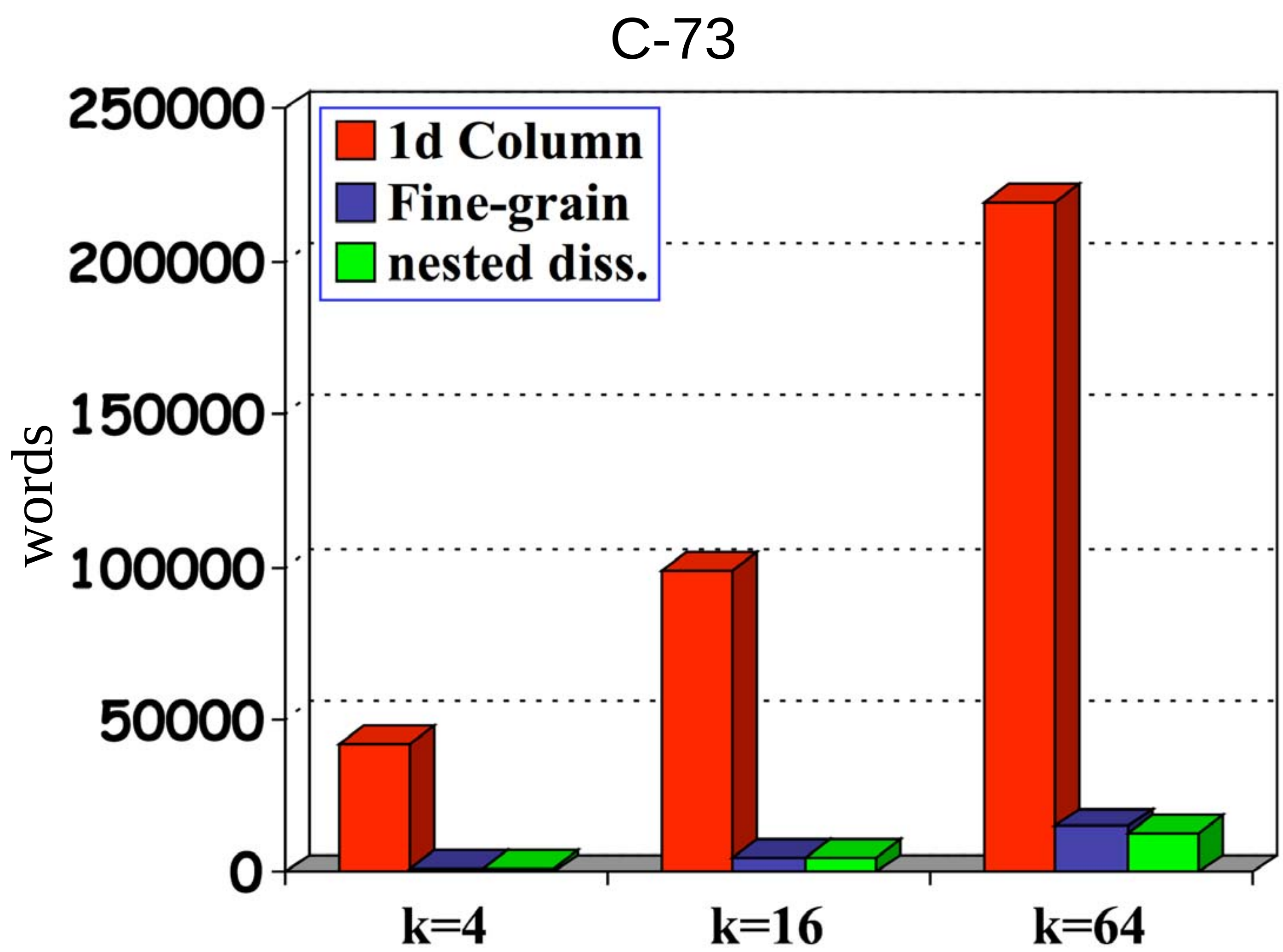


Square

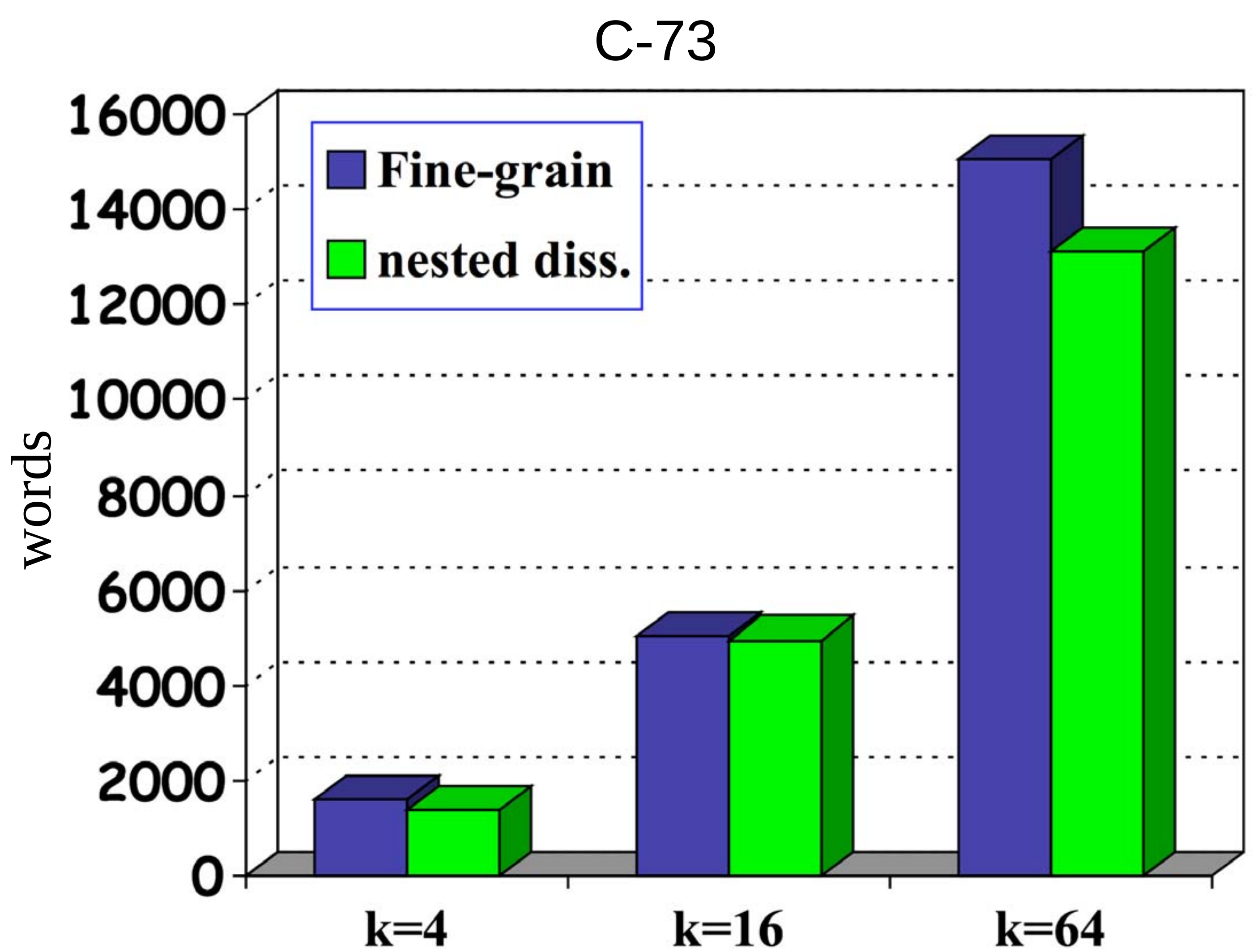




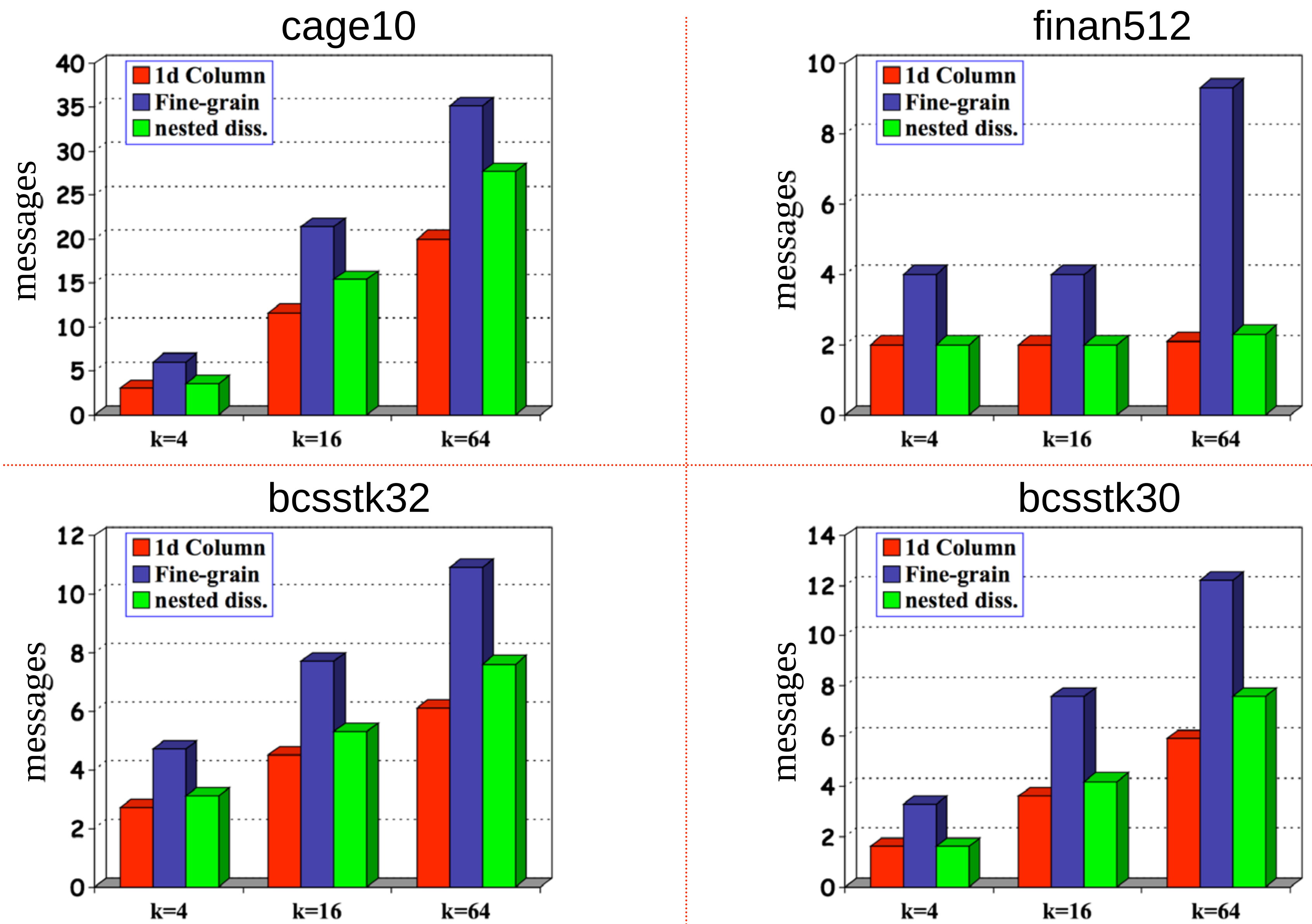
Communication Volume: 1-D is Inadequate



Communication Volume: 1-D is Inadequate



Messages Sent (or Received) per Process



Conclusions

- Several ways to reduce communication in sparse matrix-vector multiplication
  - Rich combinatorial problem!
- 1-D matrix partitioning
  - Works well for many problems (meshes)
  - Insufficient for many more irregular matrices
- New nested dissection 2-D algorithm
  - Implemented using existing algorithms and software
  - Quality better than 1-D, and similar to fine-grain hypergraph method for many matrices
  - Faster to compute than fine-grain hypergraph
  - Fewer messages than fine-grain hypergraph