EE382V (17325): Principles in Computer Architecture
Parallelism and Locality
Fall 2007

Lecture 24 – Lab 2 Review + Increasing Locality

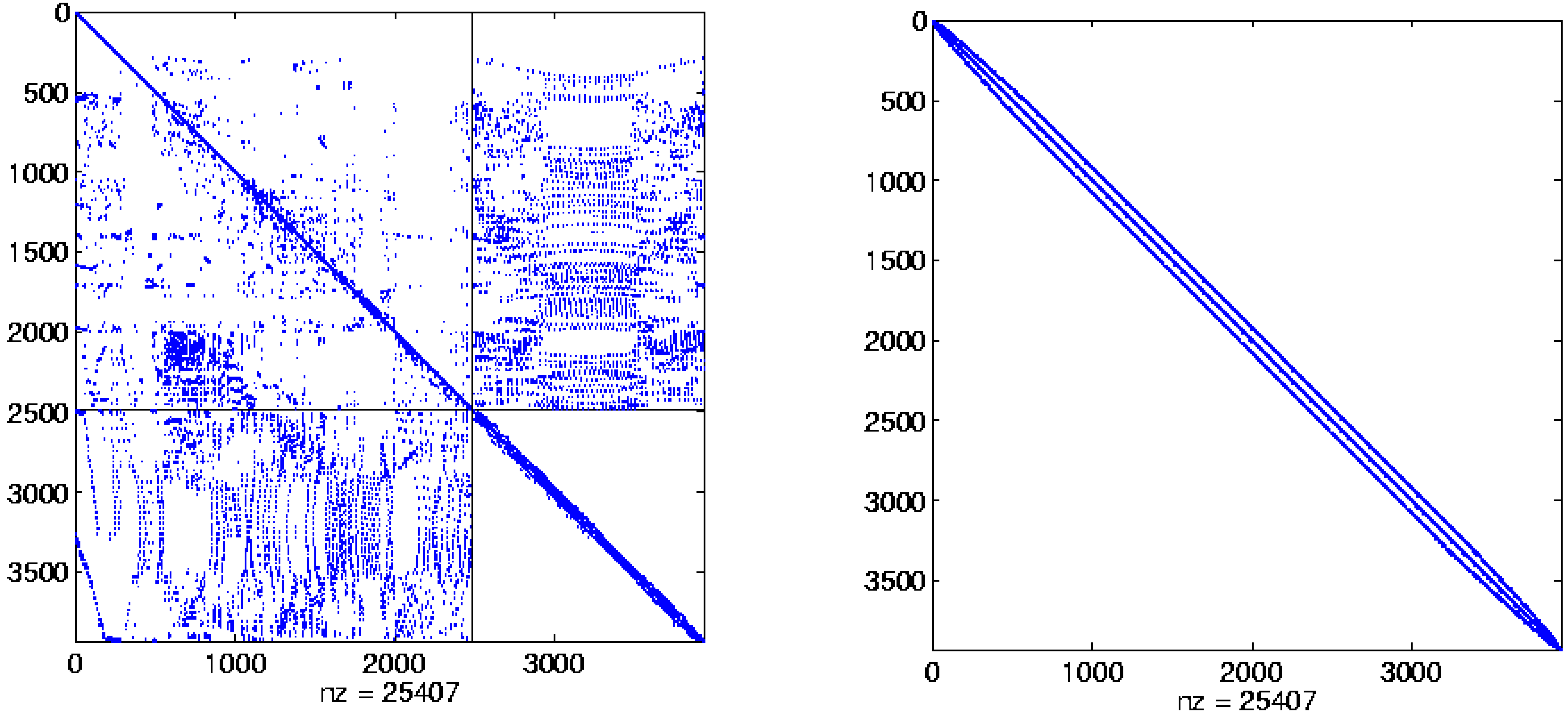
Mattan Erez

The University of Texas at Austin

Outline

- Lab 2 Review
- Overview of locality increasing tools
  - Data partitioning / domain decomposition
- Examples
  - From many sources on the web (quoted on relevant pages)

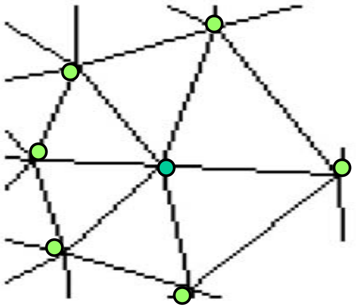
Sparse Matrix Example

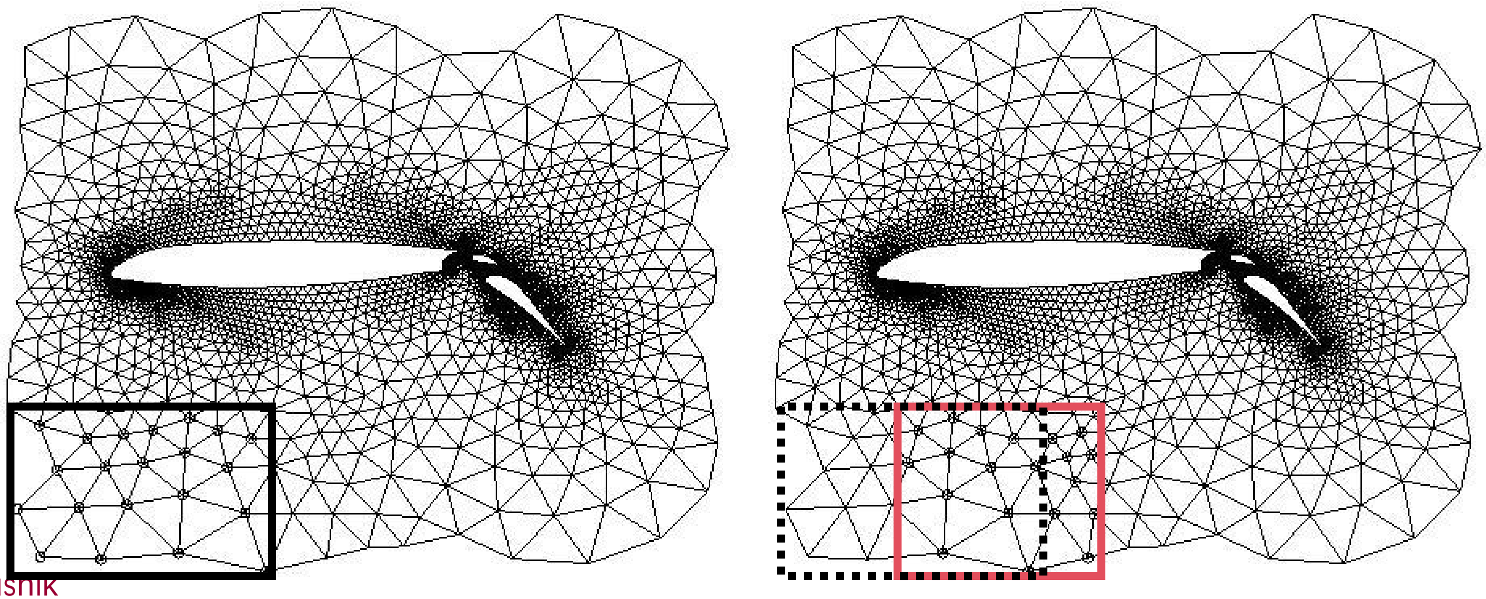


© http://www.math.tu-berlin.de/ilupack/doc/ilupack.html

Locality in PDEs

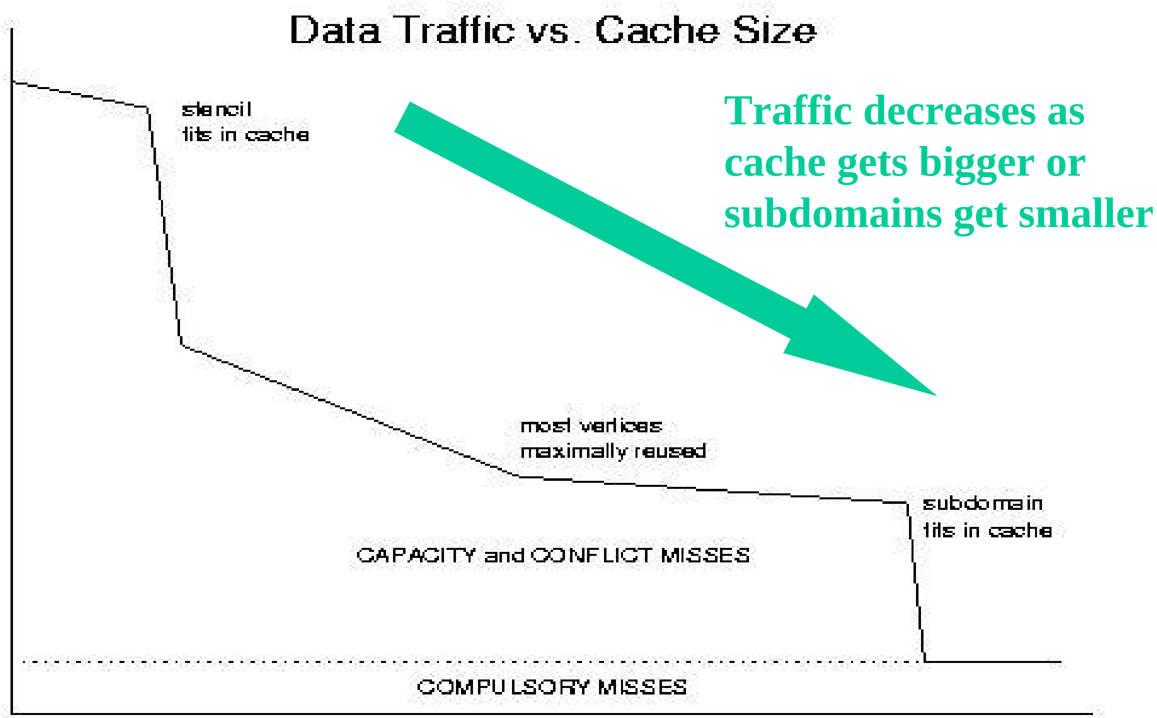
- Smallest: data for single stencil
- Largest: data for entire subdomain
- Intermediate: data for a neighborhood collection of stencils, reused as possible



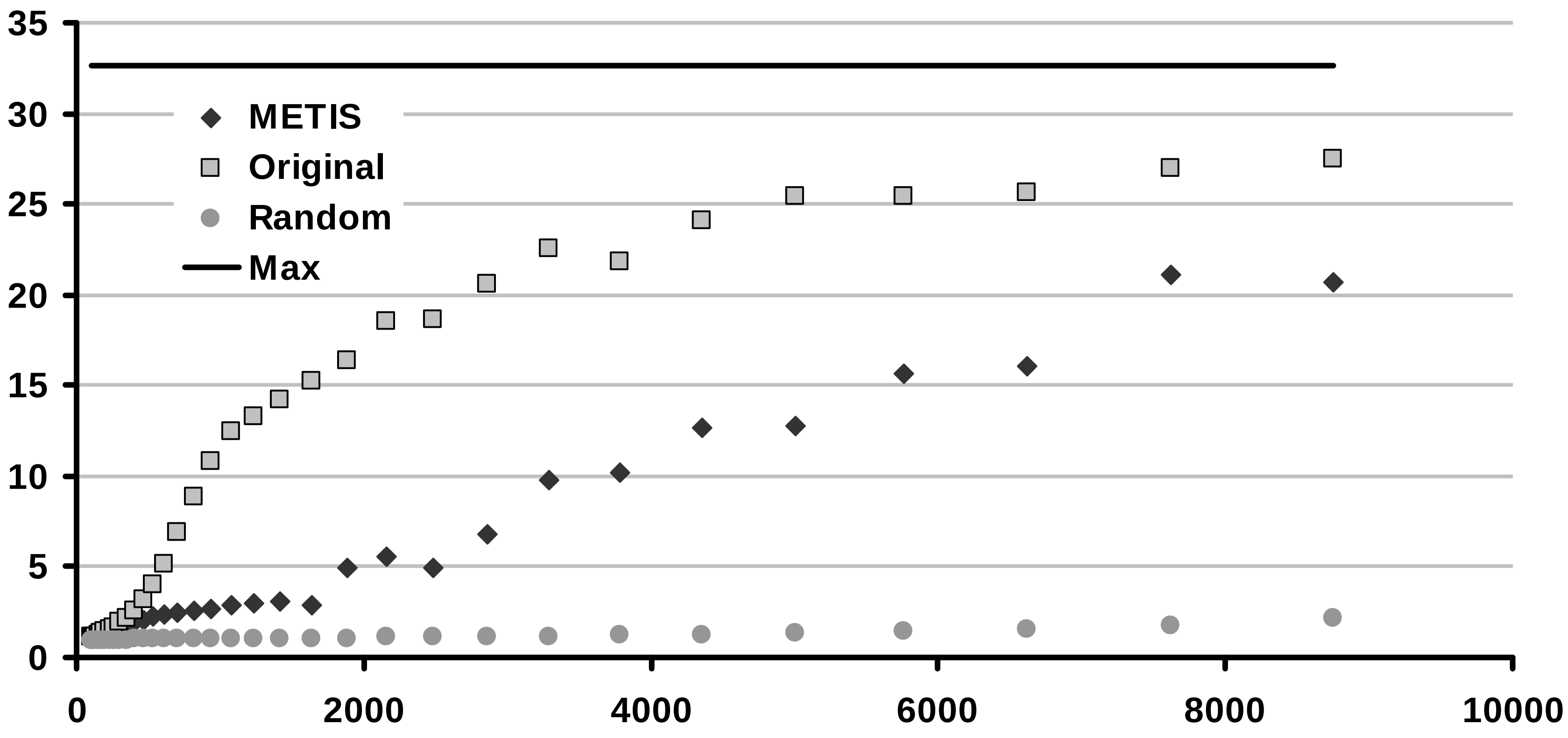


Locality in PDEs

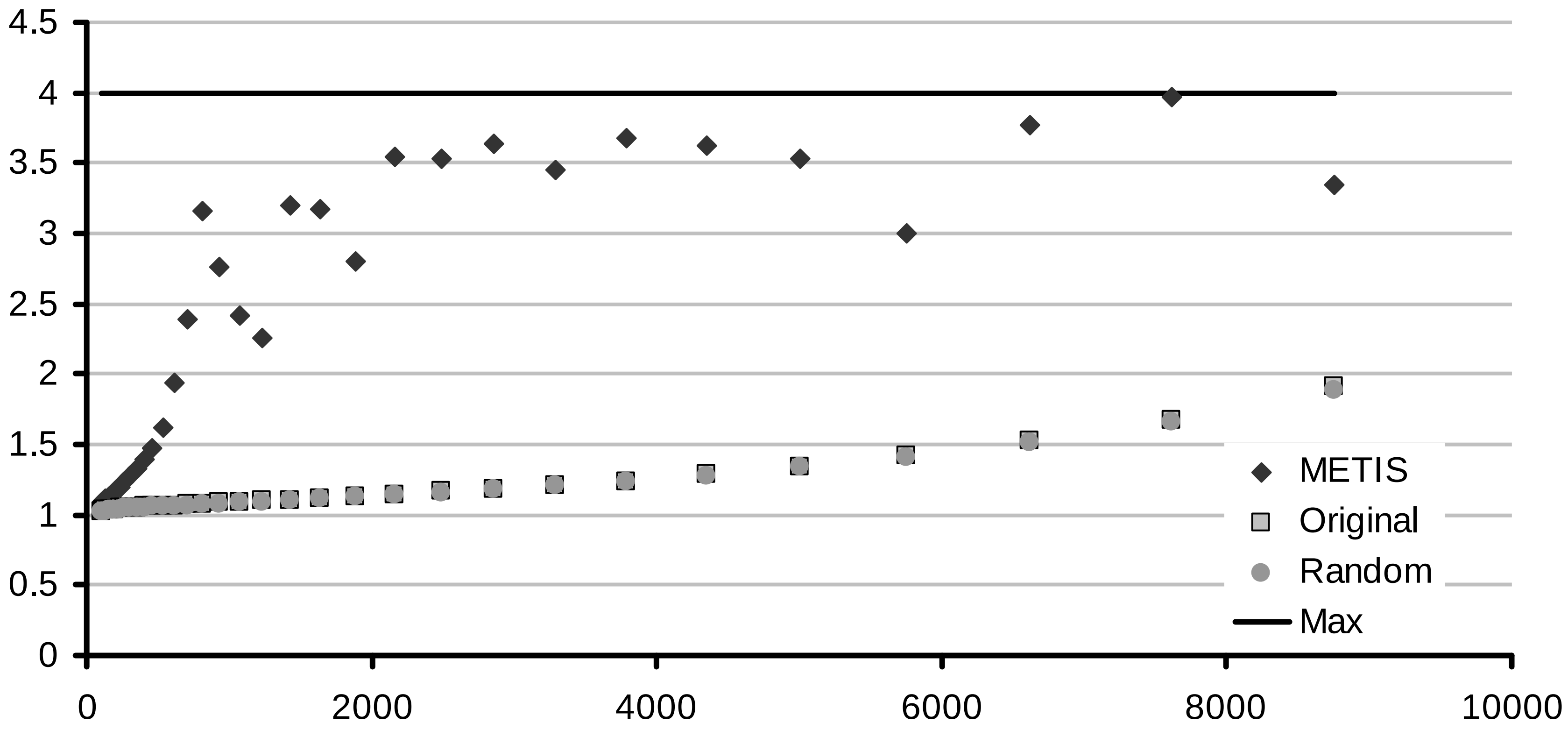
- As successive workingsets “drop” into a level of memory, capacity (and with effort conflict) misses disappear, leaving only compulsory, reducing demand on main memory bandwidth



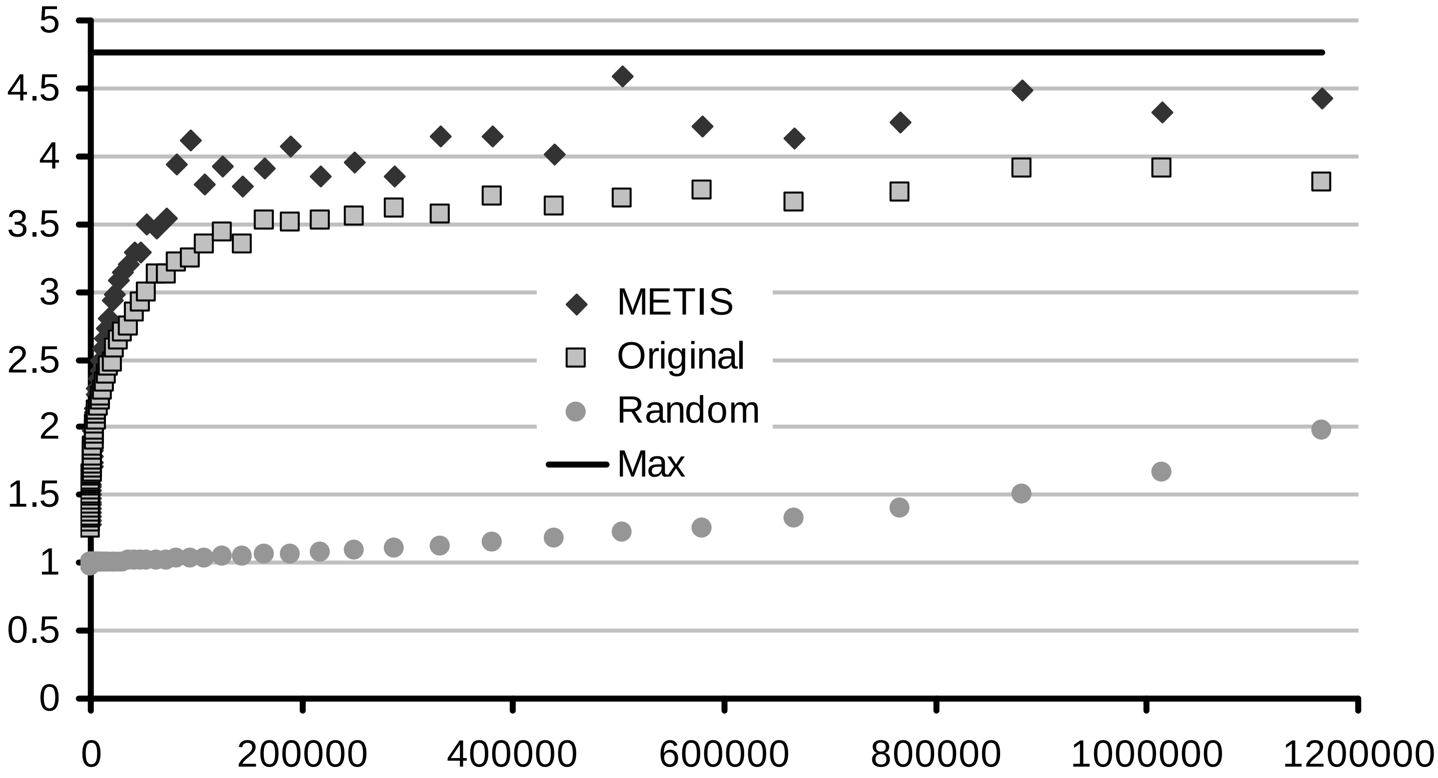
Reordering of Water Molecules



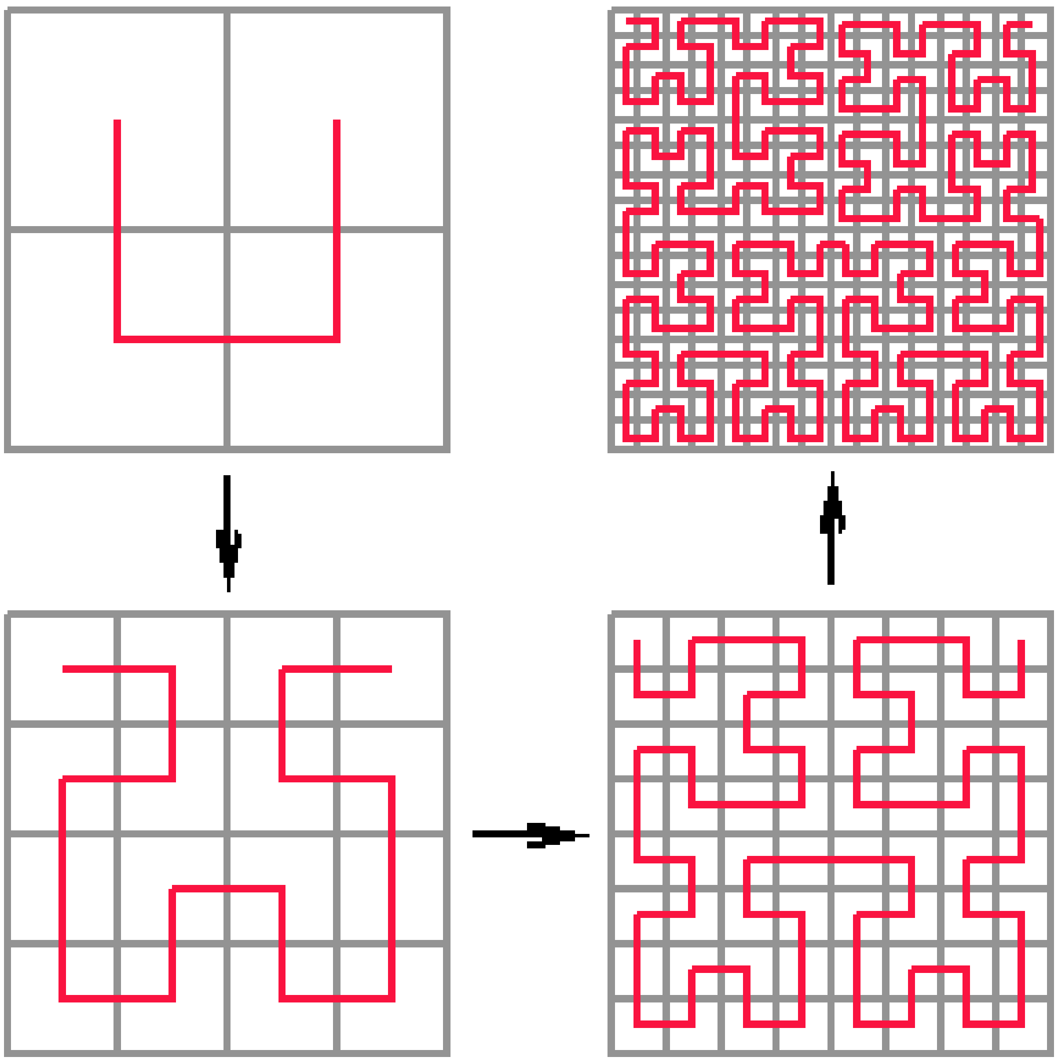
Reordering of Finite Element Mesh



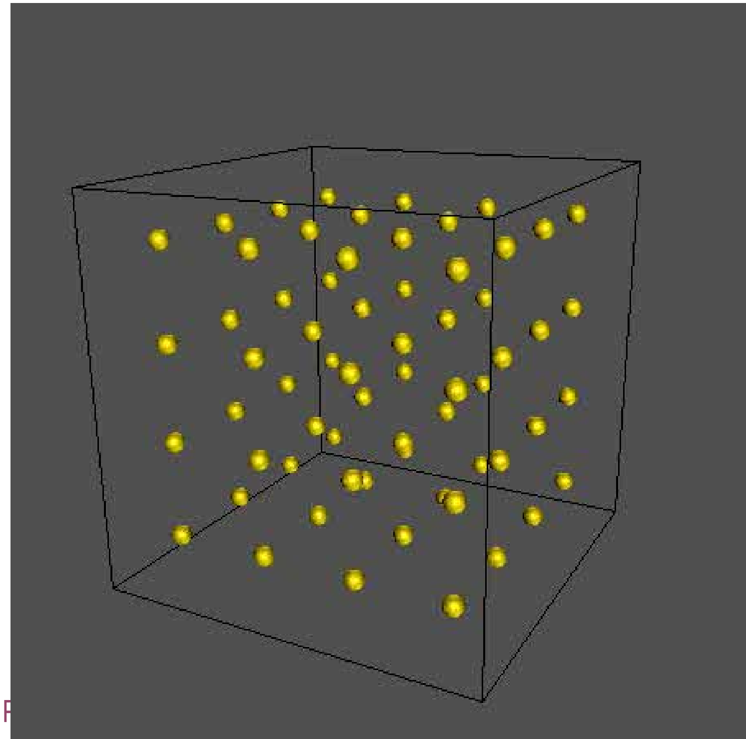
Reordering of Finite Element Mesh



Space Filling Curves

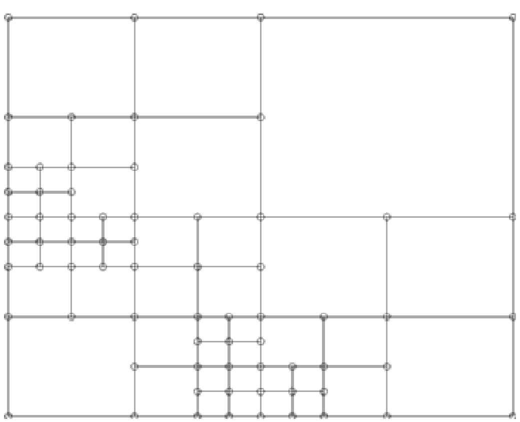


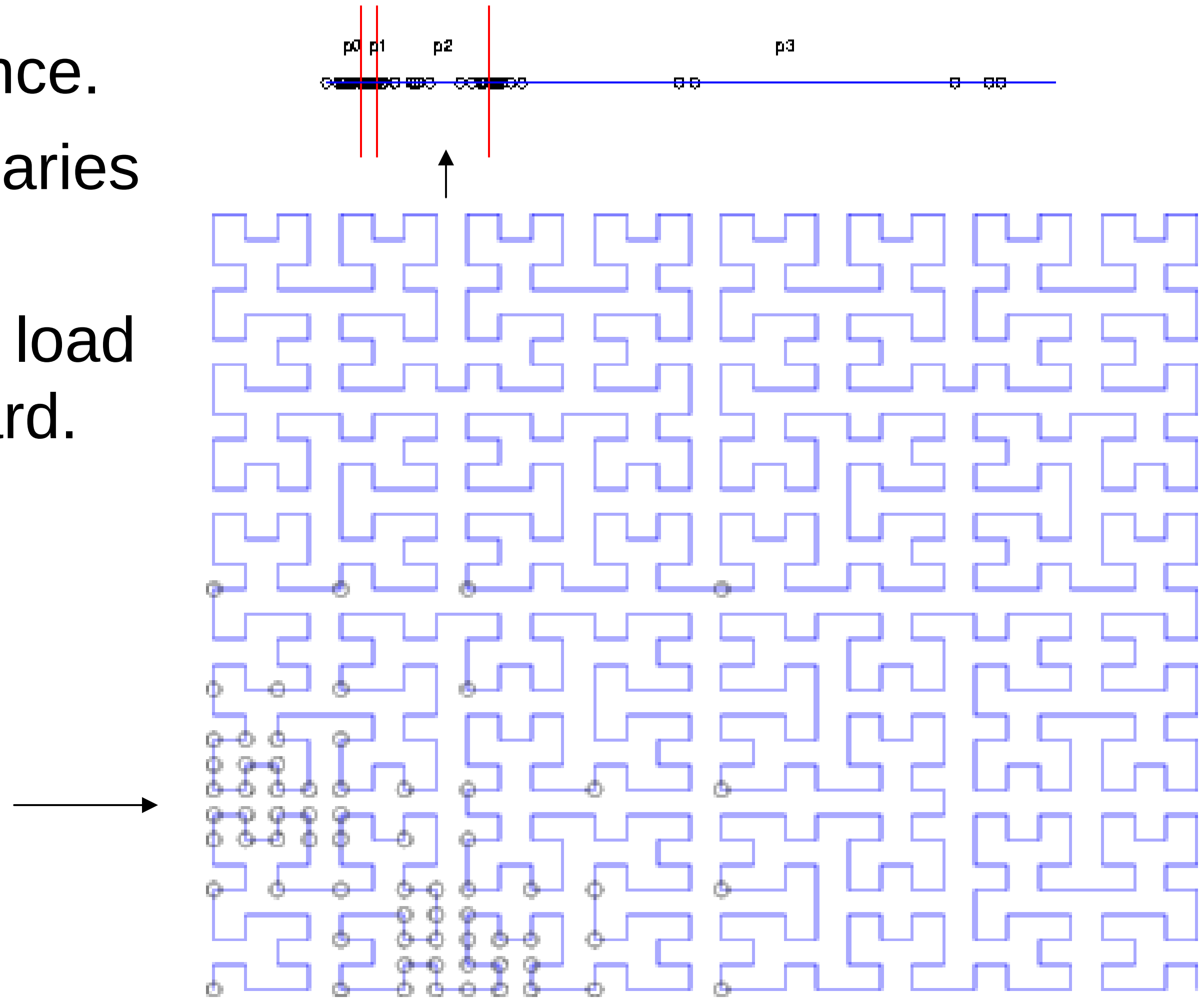
- Locality preserving.
- Each point lies a unique distance along the curve.



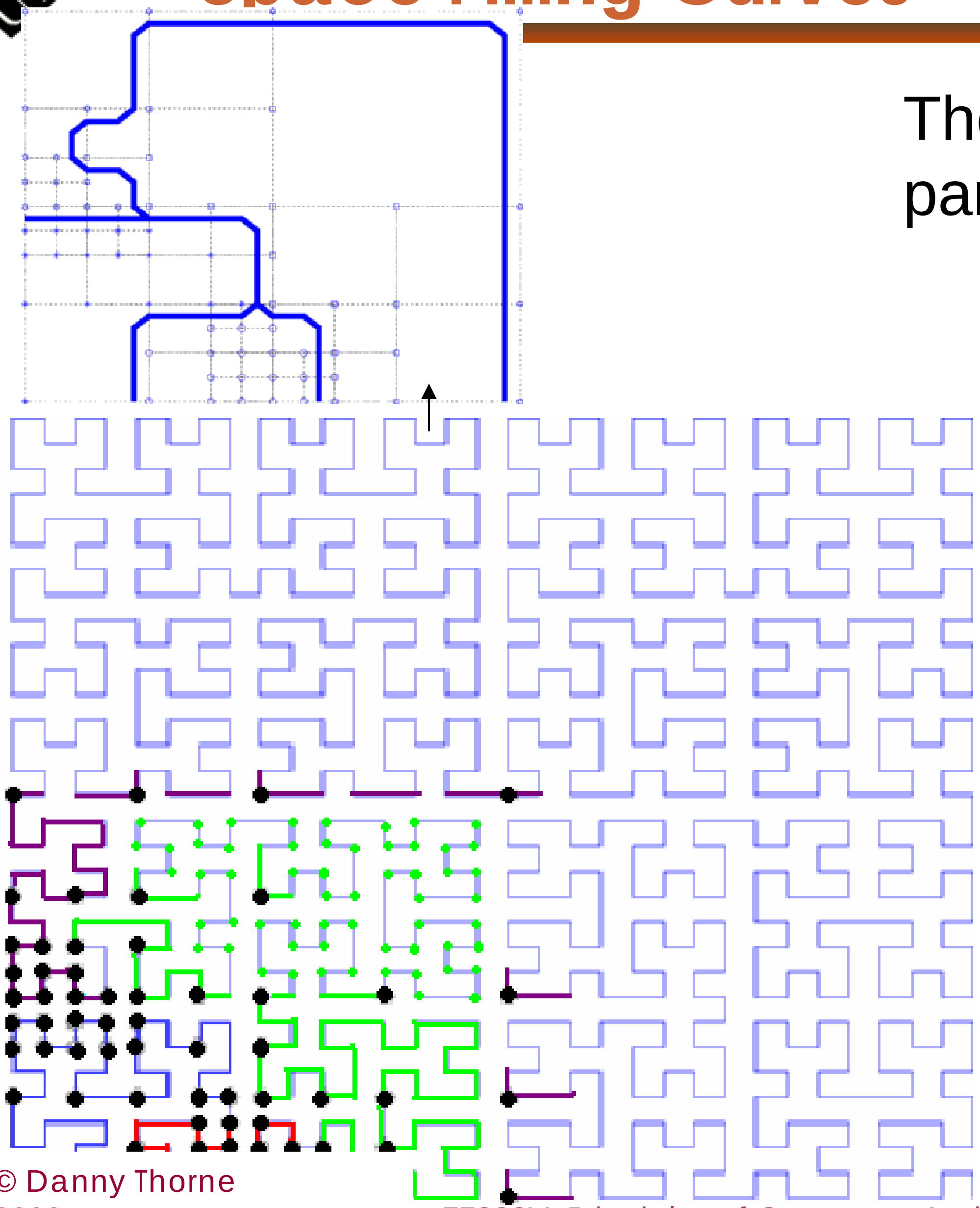
Space Filling Curves

- Optimal load balance.
- Subdomain boundaries are sub-optimal.
- Recall: Optimizing load and comm is NP-hard.





Space Filling Curves



The main advantages of this partition method are:

- It is fast compared to graph partitioning heuristics,
- It runs in parallel,
- It requires no administration and no storage of processor neighborhoods.
- The knowledge of the separators is enough to compute where to find a node and which processor to ask for it.

Streaming Meshes

- Various reorderings for a mesh for streaming

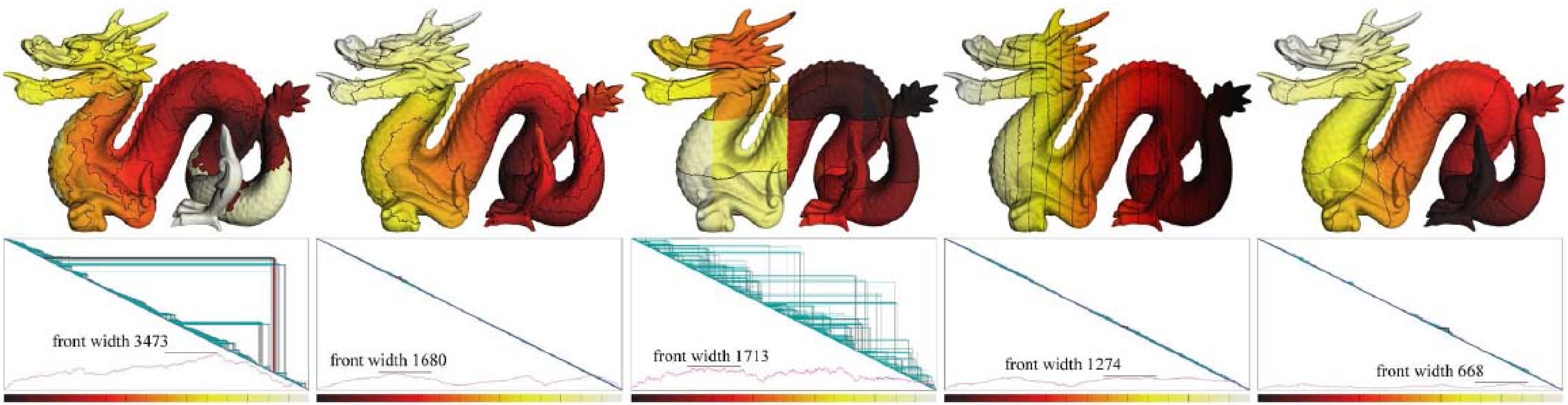
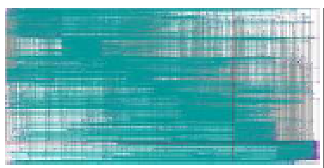
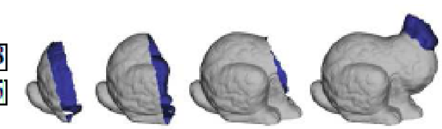
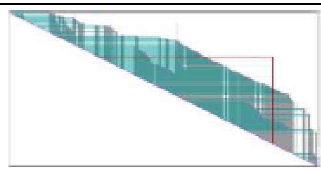
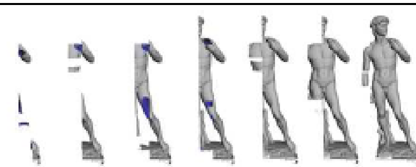
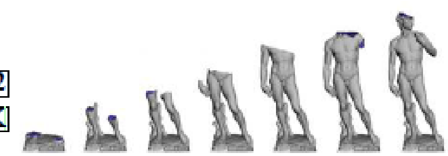
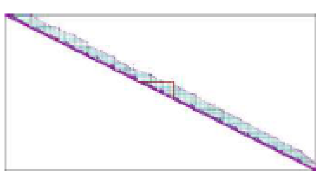
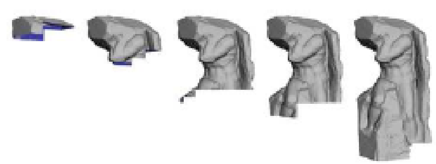
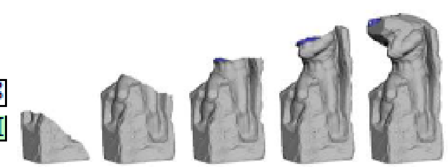


Figure 6: The dragon mesh reordered by (a) a depth-first compressor, (b) a breadth-first compressor, (c) z-order curve, (d) spatial sort, and (e) spectral sequencing.

From: Martin Isenburg and Peter Lindstrum, “Streaming Meshes”, 2005

Streaming Meshes

mesh description name genus # comp. # vertices # triangles	 skip v-width t-width v-span t-span	layout diagram	original layout *inter-leaved* width span	 *v-com-pacted* width span	 *t-com-pacted* width span	 snapshots	spectral sequencing width span	 snapshots
bunny 0 1 35,947 69,451	34,569 9,133 11,135 35,742 69,181		34,813 34,834	9,133 34,549	3,924 34,641		228 785	
david_{1mm} 137 2,322 28,184,526 56,230,343	1,568 26,383 52,515 15.8M 31.5M		26,405 15.8M	26,383 15.8M	26,375 15.8M		7,862 752K	
atlas 5,496 38 254,837,027 507,512,682	139 28,701 58,281 30.6M 61.2M		28,705 30.6M	28,701 30.6M	28,701 30.6M		45,998 28.5M	

From: Martin Isenburg and Peter Lindstrum, "Streaming Meshes", 2005