Load Distribution in MADNESS

Rebecca J. Hartman-Baker
Computer Science and Mathematics Division
Oak Ridge National Laboratory*
hartmanbakrj@ornl.gov
http://www.csm.ornl.gov/~hqi/

Computational Science & Engineering Seminar
University of Illinois at Urbana-Champaign
August 6, 2007

*Managed by UT Battelle, LLC under Contract No. DE-AC05-00OR22725 for the U.S. Department of Energy.

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

1

Introduction

I. MADNESS
II. Load Balancing
III. Load Balancing in MADNESS
IV. Implementation and Results
V. Further Work

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

2

I. MADNESS

- About MADNESS
- Computing goals
- Motivation for this work
- Math overview

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

3

MADNESS

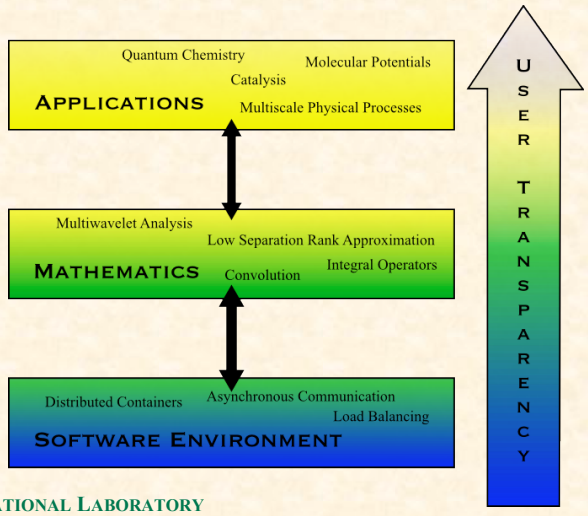
- **M**ultiresolution **Ad**aptive **N**umerical **E**nvironment for **S**cientific **S**imulation
- "Environment for prototyping and developing scientific applications using multiresolution analysis and low separation rank methods"
- Project participants include George Fann, Robert Harrison, Gregory Beylkin, Shinichiro Sugiki, and Arianna Beste

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

4

MADNESS



OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

5

Computing Goals

- Run on leadership supercomputers; e.g., ORNL's Jaguar Cray XT-4, Argonne's BlueGene/L, more advanced future systems
- Compute using 10^3–10^4 processors
- Solve problems having trees with millions or billions of nodes

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

6

Motivation for this Work

- Effective use of leadership resources requires even distribution of computational work
- Efficiency suffers if load imbalanced
- Implementing good load balancing techniques crucial to development of MADNESS

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

7

Math Overview

- Multiresolution analysis
- Low separation rank methods

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

8

Multiresolution Analysis: Representing a Function

- **Fourier representation:** $f(x) = a_0 + \sum_{k=1}^{\infty}(a_k \sin kx + b_k \cos kx)$
  - **Any periodic function breaks down into sum of sines and cosines**
  - **Drawbacks: sinusoids very smooth, very broad**
  - **Poor performance for representation of non-smooth functions (e.g., Gibbs phenomenon)**
- **Wavelet representation:**
  - **Basis generated by dilating and translating single scaling function $\psi(x)$**
  - **Subject to good choice of $\psi(x)$, any L^2 function can be approximated up to arbitrarily high precision**

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

9

Multiresolution Analysis: Scaling Function Space

- **Chain of embedded, closed subspaces**

$$\cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \cdots$$

- **with properties**

$$L^2([0,1]) = \overline{\bigcup_j V_j} \quad \text{and} \quad \bigcap V_j = \{0\}$$

- **Subspace V_0 invariant under integer translations**
- **Subspaces V_j all scaled versions of one another**
- **One or more scaling functions are in V_0 such that rescaled and shifted versions, of form $2^{j/2}\phi(2^j x-k)$, constitute orthonormal basis of V_j.**

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

10

Low Separation Rank (LSR) Methods

- Separation rank:
  - Generalization of separation of variables
  - Value m such that for d-dimensional function

$$\left| f(x_1, x_2, \ldots, x_d) - \sum_{i=1}^{m} s_i \phi_1^i(x_1) \phi_2^i(x_2) \cdots \phi_d^i(x_d) \right| < \varepsilon$$

- Benefits of LSR representations
  - All operations are one-dimensional
  - Low storage requirements
  - Constructive algorithms for reduction of separation rank exist
  - Removes curse of dimensionality
- Many "interesting"operators and functions have LSR representations

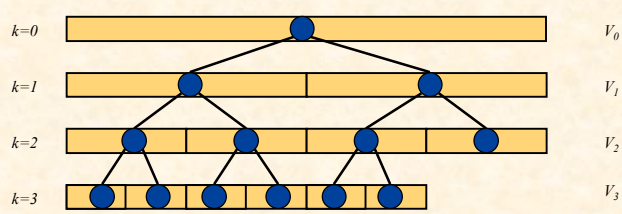
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

11

Representing a Function

- Telescoping subspaces V_k translate into tree form



- Adaptive refinement and truncation of coefficients means not all intervals subdivided

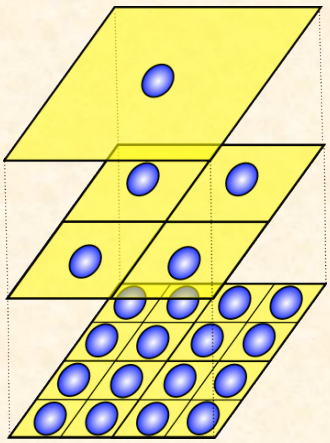
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

12

Representing a Function

- Same form for multiple dimensions (tensor product)
- Typical basis functions: Legendre multiwavelet basis (Legendre polynomials rescaled to unit interval)



OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

13

Trees

- Result of adaptive refinement of spatial domain
  - 2-D: quadtrees
  - 3-D: octrees
- $O(10^6)$–$O(10^9)$ nodes

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

14

II. Load Balancing

- Load balancing paradigms
  - Overview
  - Pros and Cons
- Existing software

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

15

Load Balancing Paradigms

- Geometric partitioning methods
- Global graph-based partitioning methods
- Local graph-based partitioning methods
- Random distribution
- (Octree partitioning methods)

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

16

Geometric Partitioning Methods

- Exploit geometric coordinates of local domain to keep data local
- Useful when data locality is important; e.g., particle simulations
- Examples: recursive coordinate bisection, space-filling curves
- Pros:
  - Easy to implement
  - Quick execution
- Cons:
  - Some methods unsuitable for dynamic load balancing
  - Produces partitions of moderate quality

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

17

Global Graph-Based Partitioning Methods

- Represent problem domain as graph and find best graph partitioning
- Can also represent domain as hypergraph, for more accurate modeling of communication costs
- Example: recursive spectral bisection, multilevel partitioning
- Pros:
  - Accurate models
  - Partition of very high quality
- Cons:
  - Very expensive to implement
  - Not readily parallelizable

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

18

Local Graph-Based Partitioning Methods

- Improve load balance over subset of load (e.g., subset of processors, subset of domain)
- Example: diffusion model of dynamic load balancing
- Useful when point imbalances occur or in case of completely embarrassingly parallel load
- Pros:
  - Easy to implement
  - Readily parallelizable
  - Fast
  - Produce good partitions
- Cons:
  - Orthogonal goals (processors vs domain)

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

19

Random Distribution

- Get good load balance by randomly distributing work over processors
- Pros:
  - Excellent balance of work
  - Trivial to implement
- Cons:
  - No locality
  - High communication volumes

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

20

Octree Partitioning Methods

- Subset of geometric partitioning methods
- Exploit octree structure of geometric partitioning
- Pros:
  - Easy to implement
  - Quick execution
- Cons:
  - Usually, for finite elements, care only about leaf nodes, but we need to partition entire tree
  - For partitioning octree structure, data locality only obliquely related to tree locality

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

21

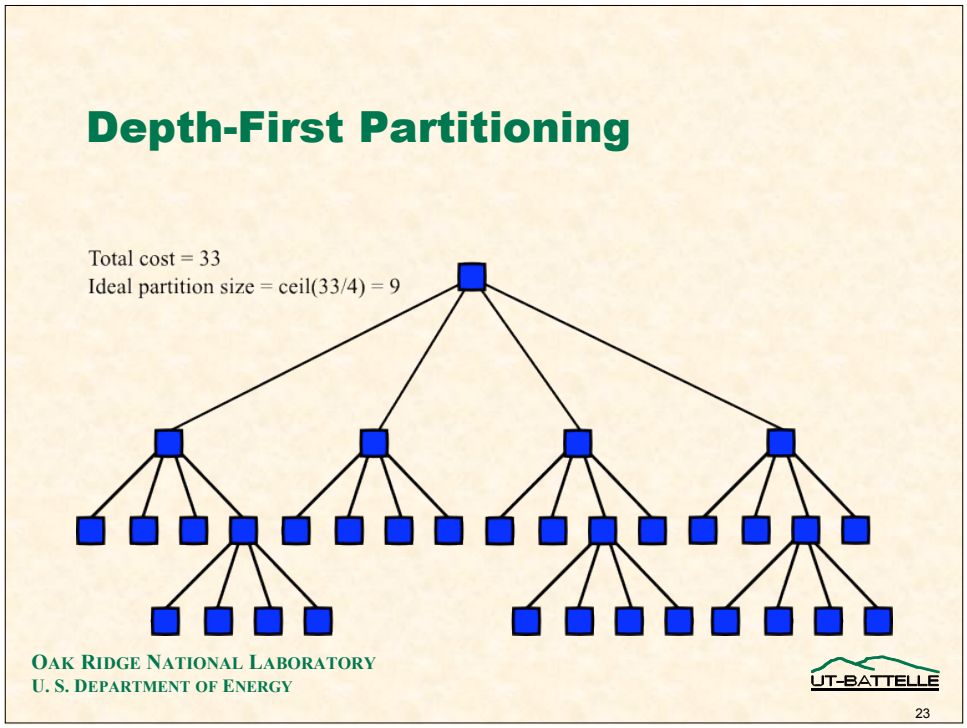
Octree Partitioning Methods

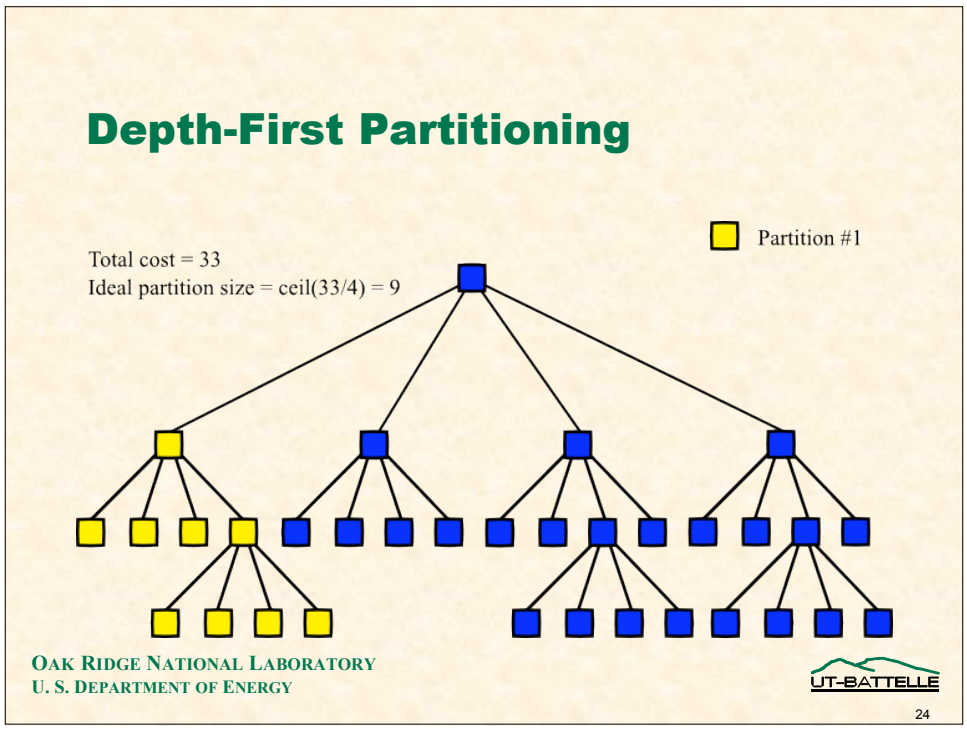
- Determine ideal partition size
- Traverse tree, adding subtrees to partition until filled, then proceed to next partition
- Continue traversing tree until all partitions full
- Traversal method:
  - Depth-first traversal
  - Space-filling curve traversal

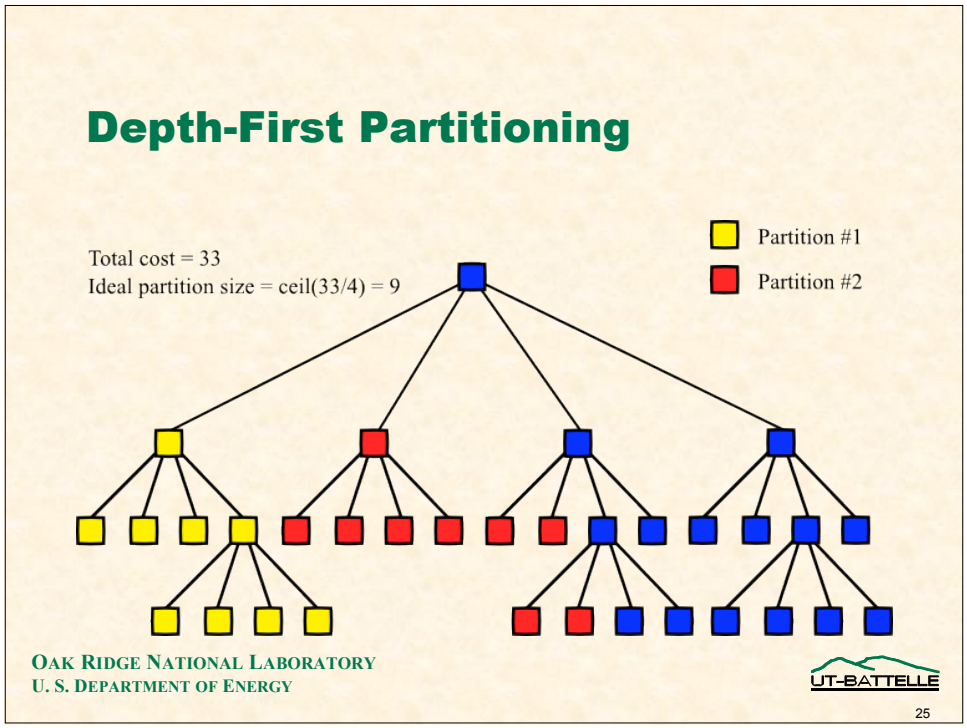
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

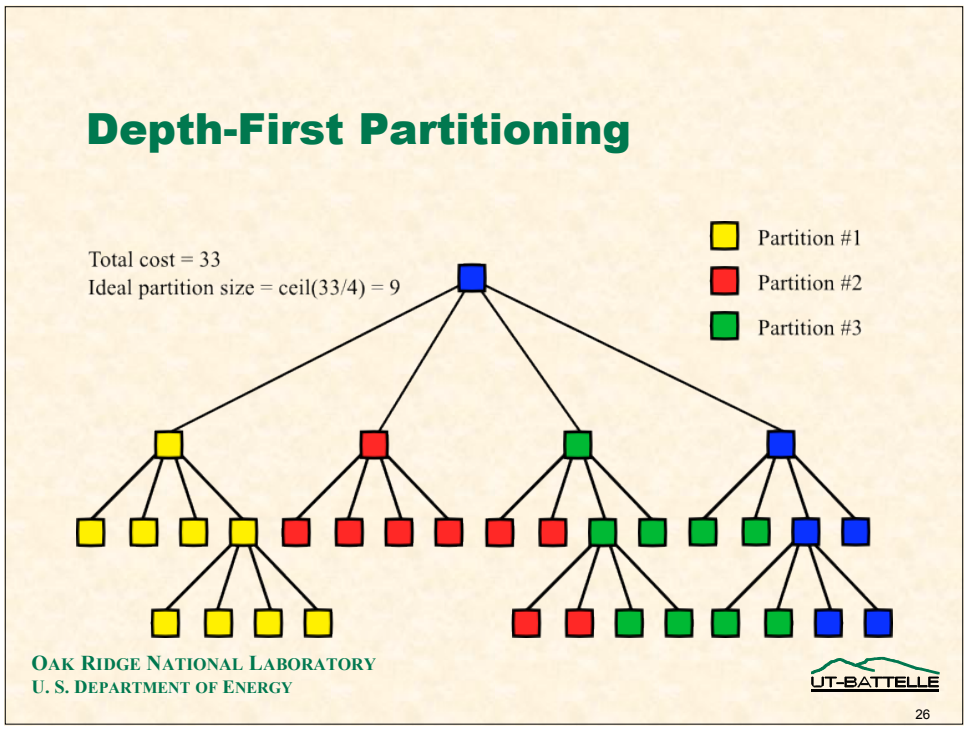
UT-BATTELLE

22


Depth-First Partitioning
Total cost = 33
Ideal partition size = ceil(33/4) = 9
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY
UT-BATTELLE
23


Depth-First Partitioning
Partition #1
Total cost = 33
Ideal partition size = ceil(33/4) = 9
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY
UT-BATTELLE
24


Depth-First Partitioning
Total cost = 33
Ideal partition size = ceil(33/4) = 9
Partition #1
Partition #2
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY
UT-BATTELLE
25


Depth-First Partitioning
Total cost = 33
Ideal partition size = ceil(33/4) = 9
Partition #1
Partition #2
Partition #3
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY
UT-BATTELLE
26

Existing Software

- Geometric Partitioning
  - Zoltan
  - DRAMA
- Graph-Based Partitioning
  - Zoltan
  - ParMetis
  - PJostle

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

27

III. Load Balancing in MADNESS

- Formal problem statement
- Motivation
- Melding heuristic
- Example

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

28

Formal Statement of Load Balancing Problem (1)

- Given a tree T with N nodes $\{t_0,\ldots,t_{N-1}\}$ and machine M with P processors $\{m_0,\ldots,m_{P-1}\}$, find mapping s.t. every $t_i \in I = \{0,\ldots,N-1\}$ is assigned to single $m_j \in J = \{0,\ldots,P-1\}$ at cost c_{ij}
- Formally: $\min \sum_{i\in I}\sum_{j\in J} c_{ij}x_{ij},$

where $x_{ij} = \begin{cases} 1 & \text{if } t_i \rightarrow m_j \\ 0 & \text{otherwise} \end{cases}$

$$\sum_{j\in J} x_{ij} = 1 \quad \forall i \in I$$

$$\sum_{i\in I} c_{ij}x_{ij} \le b_j \quad \forall j \in J$$

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

29

Formal Statement of Load Balancing Problem (2)

- Cost function c_{ij} is not constant
  - $c_{ij} = \alpha_i + \beta_i(1-x_{i'j})$, where $t_{i'}$ is t_i's parent
- If t_i assigned to same processor as parent, then no communication cost β_i.
- Problem is identical to Generalized Assignment Problem
  - Integer programming problem
  - NP-complete

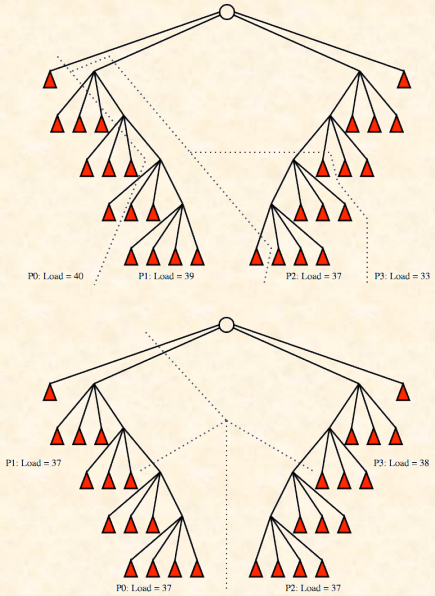
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

30

Motivation

- *Top picture:* depth-first partitioning
- *Bottom picture:* "eyeball" partitioning
- Come up with method that exploits structure of octree
- Select best partition as function of target computer's computation and communication speeds



OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

31

Melding Heuristic

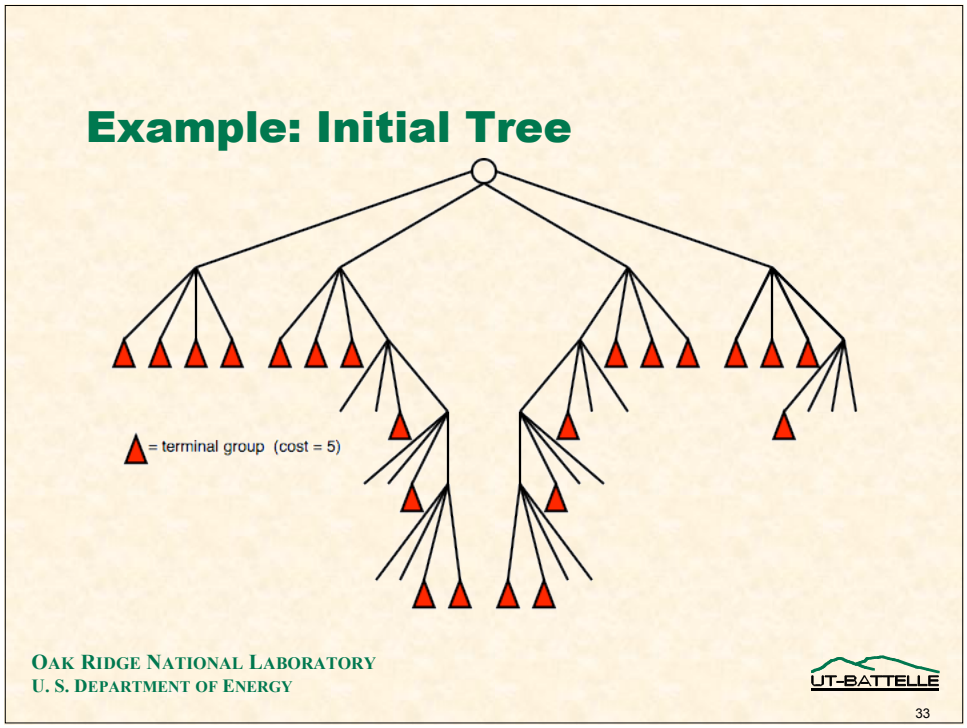
- *Melding:* make parent and leaf children inseparable before load balancing performed
- Recursively meld and load balance, and select best configuration for target computer, based on its computation and communication speeds

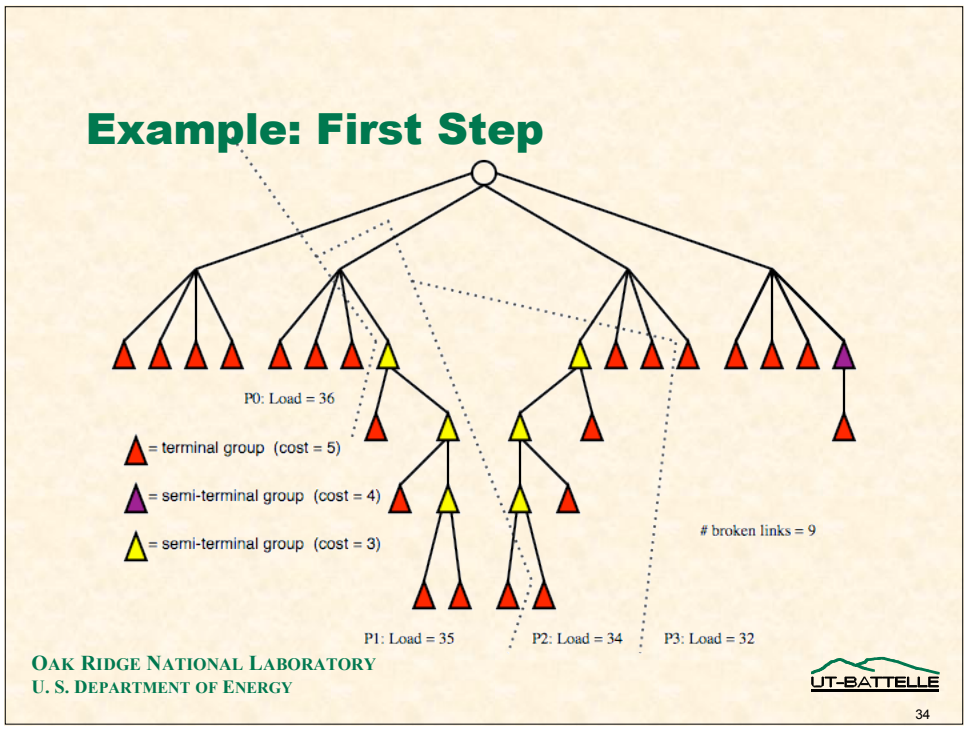
(from http://www.answers.com/topic/vulcan-star-trek)

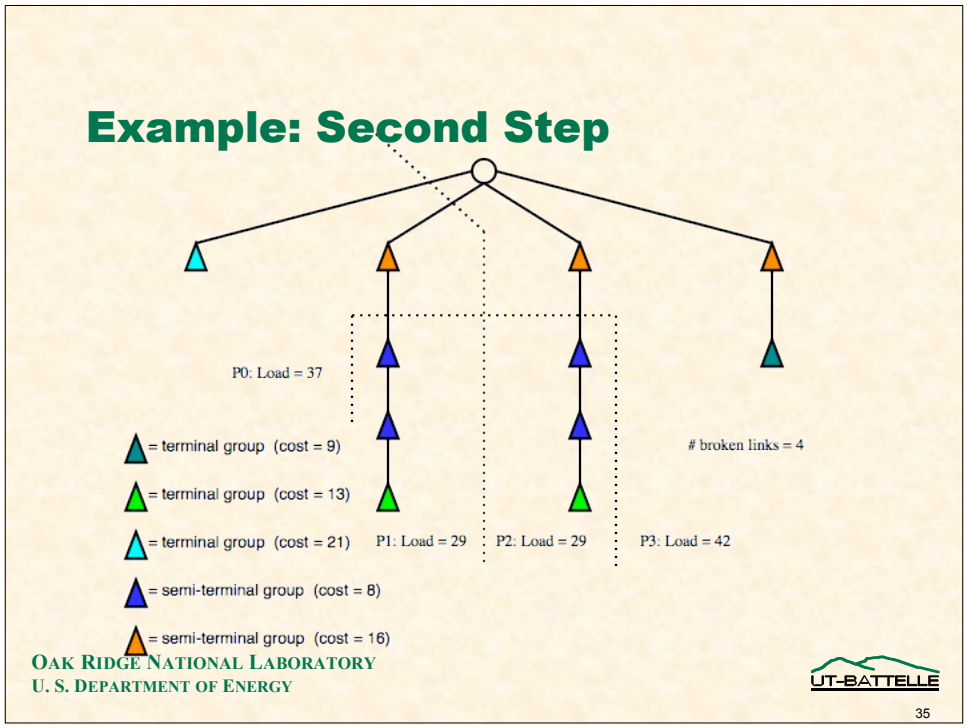
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

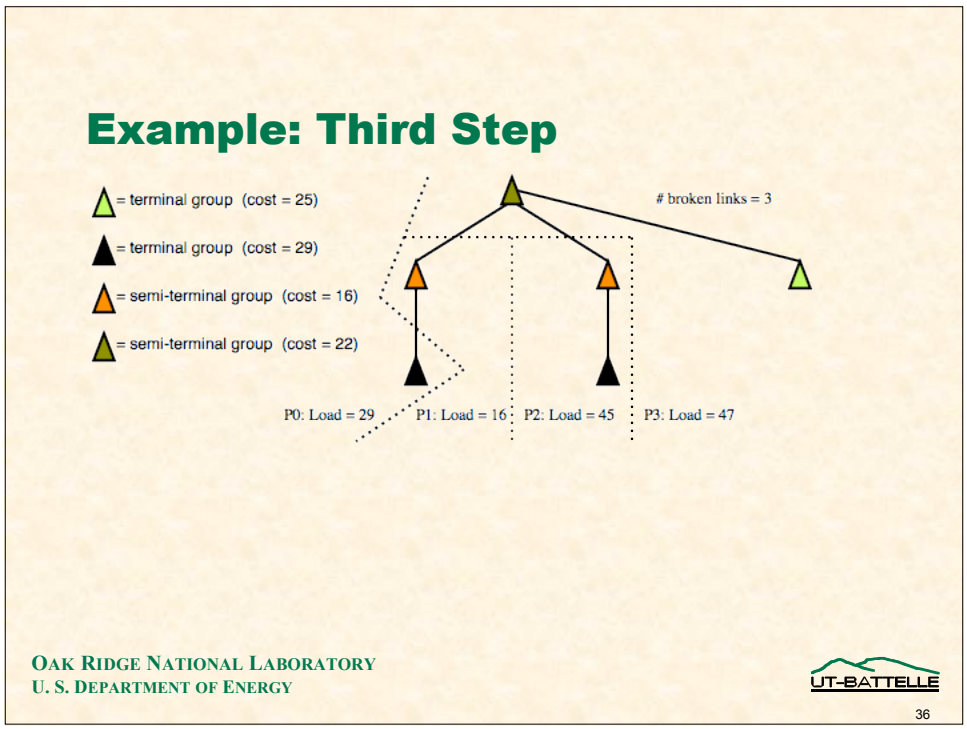
UT-BATTELLE

32


Example: Initial Tree
= terminal group (cost = 5)
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY
UT-BATTELLE
33


Example: First Step
P0: Load = 36
= terminal group (cost = 5)
= semi-terminal group (cost = 4)
= semi-terminal group (cost = 3)
broken links = 9
P1: Load = 35
P2: Load = 34
P3: Load = 32
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY
UT-BATTELLE
34


Example: Second Step
P0: Load = 37
= terminal group (cost = 9)
= terminal group (cost = 13)
= terminal group (cost = 21)
= semi-terminal group (cost = 8)
= semi-terminal group (cost = 16)
P1: Load = 29
P2: Load = 29
P3: Load = 42
broken links = 4
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY
UT-BATTELLE
35


Example: Third Step
= terminal group (cost = 25)
= terminal group (cost = 29)
= semi-terminal group (cost = 16)
= semi-terminal group (cost = 22)
broken links = 3
P0: Load = 29
P1: Load = 16
P2: Load = 45
P3: Load = 47
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY
UT-BATTELLE
36

Example: Summary

Maximum Load	# Broken Links	Cost (0.35 x Load + Links)
36	9	21.60
42	4	18.70
47	3	19.45

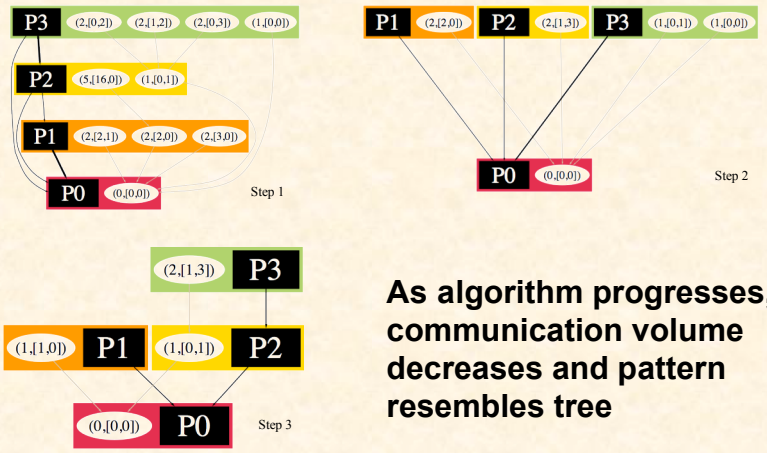
Best configuration for given machine may be suboptimal in load and broken links

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

37

Example: Communication Patterns



As algorithm progresses, communication volume decreases and pattern resembles tree

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

38

IV. Implementation and Results

- Implementation of MADNESS
- Implementation of load balancing
- Preliminary performance evaluation

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

39

Implementation of MADNESS

- Original proof-of-concept was written in Python
- New MADNESS written in C++
- Object-oriented design
- User transparency

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

from http://xkcd.com/c292.html

UT-BATTELLE

40

Implementation: Trees

- Trees stored across machine in distributed container class, similar to STAPL
  - STL map-like `WorldContainer` stores node coefficients using node coordinates as key
  - Remote or local `find(key)` transparent to user
- Procmap: assigns nodes to processors based on
  - Hashing of key (`hash(key) mod nproc`)
  - Location of nearest ancestor

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

41

Implementation: Communication

- Multithreaded parallel asynchronous communication (Cilk-like)
- Asynchronous communication with Futures
  - Store unevaluated expression in variable of type Future; reference to result forces evaluation of Future if not yet evaluated
  - Implemented as templated C++ class with thread pool

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

42

Implementation: Load Balancing

- Load Balancing class `LoadBalImpl` sets up load balancing
- Creates stand-in (skeleton) tree (in another `WorldContainer)` having same structure as octree, upon which melding algorithm is performed
- Skeleton has diagnostic data in nodes, used for depth-first partitioning
- `skel_tree.load_balance()` results in new procmap, then real data copied over to new configuration

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

43

Implementation: Depth-First Partitioning

- Like algorithm demonstrated earlier
- Assign "fudge factor," percentage for acceptable overfill
- Require method to accept single node that overfills partition+fudge factor, if node has no children and partition is empty
- Reconsider ideal partition size at beginning of each partition
  - Raise partition size if previous partitions underfilled
  - Shrink partition size if exceeds projected ideal partition size for remaining partitions by percentage

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

44

Preliminary Performance Evaluation

- Comparison of two process map paradigms
  - Keys hashed and assigned to processors based on `hash mod NP` (f)
  - Assignment to processors based on result of melding algorithm (g)
- Compare performance of
  - Compress
  - Reconstruct

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

45

Preliminary Performance Evaluation

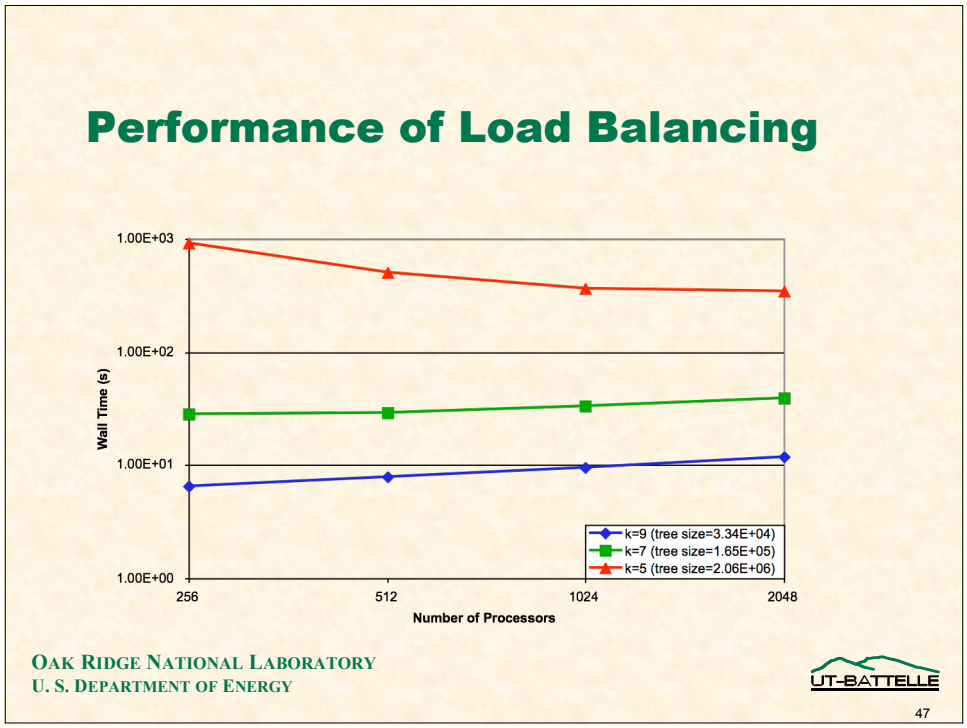
- Approximate sum of Gaussians, with accuracy threshold 10^{-12}
- Approximate using different degrees for multiwavelets: k = 9, 7, 5

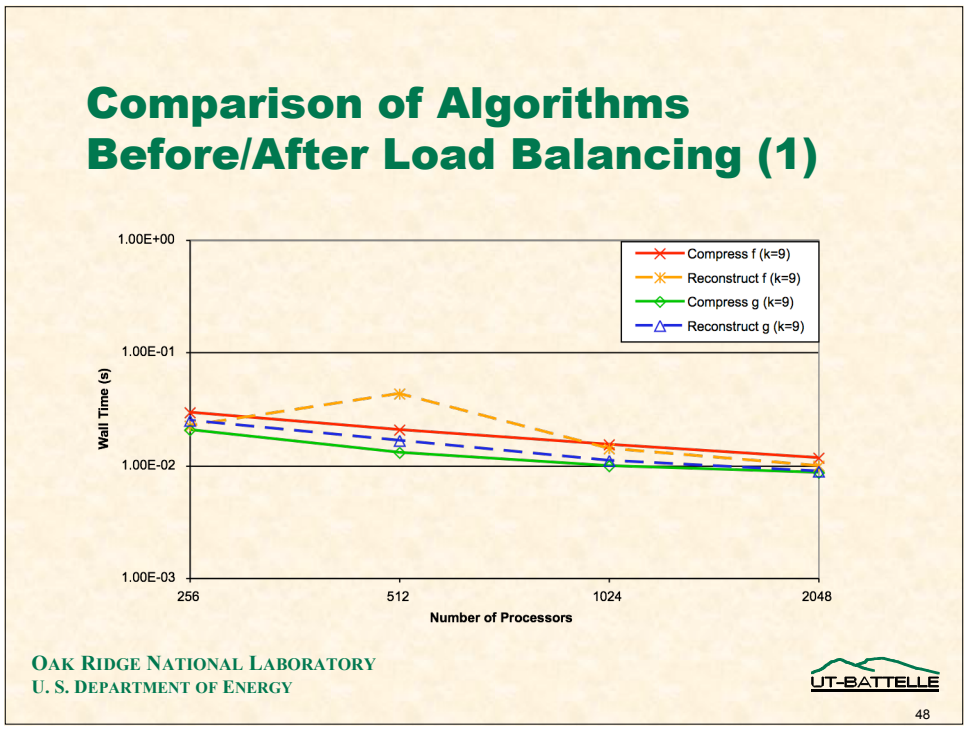
k	# Coefficients	Tree Size
9	2.13×10^{7}	3.34×10^{4}
7	4.97×10^{7}	1.65×10^{5}
5	2.25×10^{8}	2.06×10^{6}

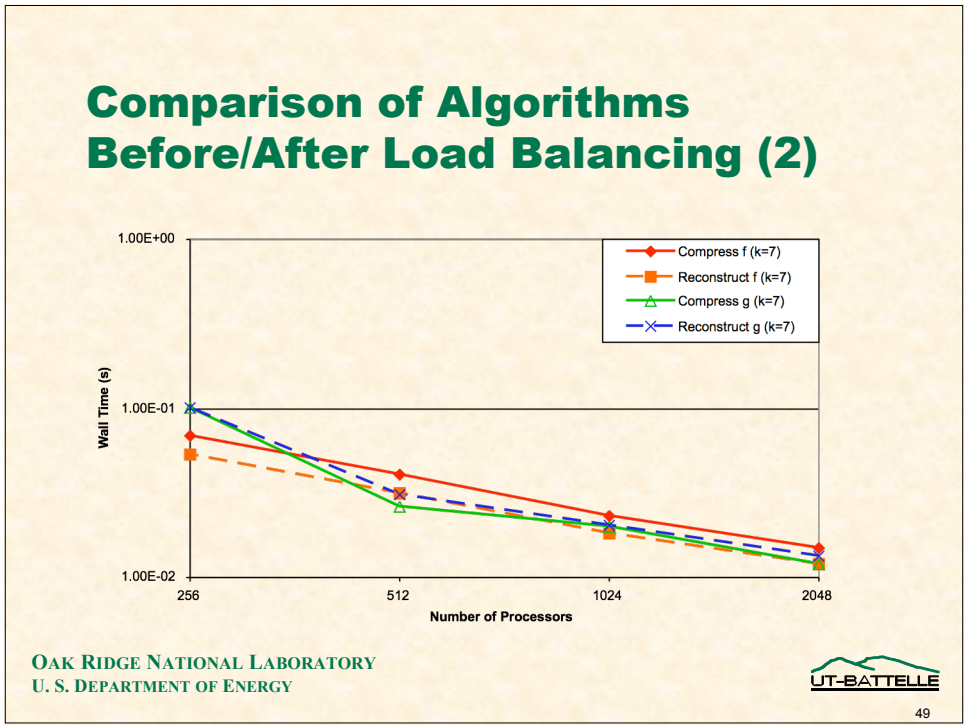
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

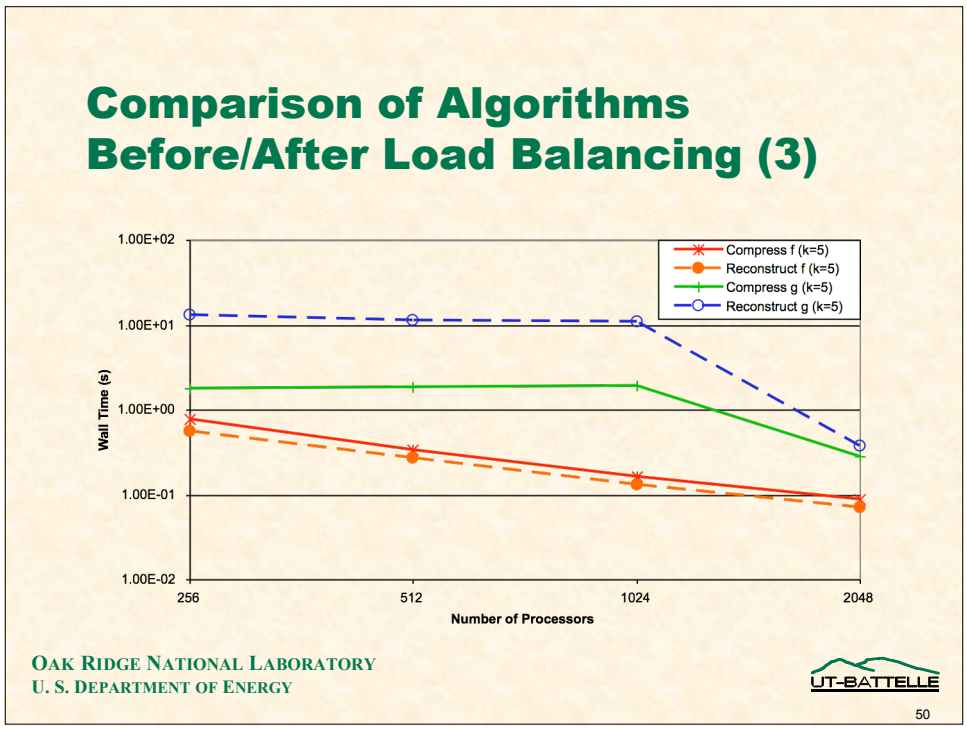
UT-BATTELLE

46

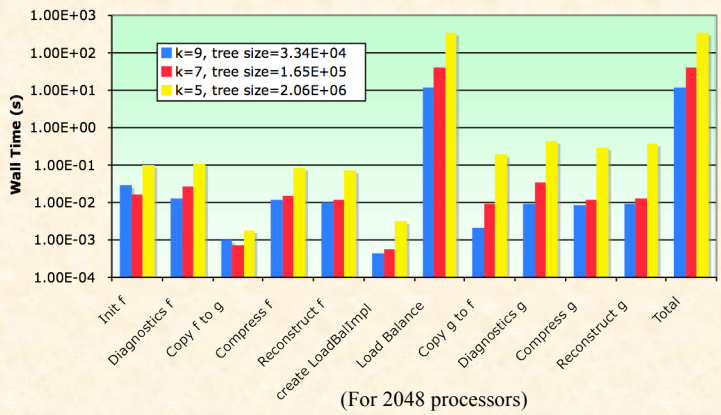

Performance of Load Balancing
1.00E+03
1.00E+02
1.00E+01
1.00E+00
Wall Time (s)
256
512
1024
2048
Number of Processors
k=9 (tree size=3.34E+04)
k=7 (tree size=1.65E+05)
k=5 (tree size=2.06E+06)
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY
UT-BATTELLE
47


Comparison of Algorithms
Before/After Load Balancing (1)
1.00E+00
1.00E-01
1.00E-02
1.00E-03
Wall Time (s)
256
512
1024
2048
Number of Processors
Compress f (k=9)
Reconstruct f (k=9)
Compress g (k=9)
Reconstruct g (k=9)
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY
UT-BATTELLE
48


Comparison of Algorithms
Before/After Load Balancing (2)
1.00E+00
1.00E-01
1.00E-02
Wall Time (s)
256
512
1024
2048
Number of Processors
Compress f (k=7)
Reconstruct f (k=7)
Compress g (k=7)
Reconstruct g (k=7)
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY
UT-BATTELLE
49


Comparison of Algorithms
Before/After Load Balancing (3)
1.00E+02
1.00E+01
1.00E+00
1.00E-01
1.00E-02
Wall Time (s)
256
512
1024
2048
Number of Processors
Compress f (k=5)
Reconstruct f (k=5)
Compress g (k=5)
Reconstruct g (k=5)
OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY
UT-BATTELLE
50

Scalability as Work Grows



(For 2048 processors)

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

51

Analysis

- Melding algorithm not scalable!
  - (Working on ways to parallelize algorithm further)
- Melding assignment outperforms hashing assignment for k=9; performance erodes as tree grows
  - Algorithm prioritizes minimization of broken links at expense of balance, can tweak to pick better configuration for this machine
  - Future optimizations should help reduce cost

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

52

V. Further Work

- Planned work
  - Streamline load balancing algorithms
  - Develop recursive melding algorithm, in which "assistant manager" processors assigned chunks of tree and subset of processors
  - Adapt melding algorithm to work with forests (so assistant managers can be assigned noncontiguous subtrees)
- If left to my own devices…
  - Explore using graph partitioning algorithms to partition melded trees
  - Further refine cost function, making more sophisticated

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

53

More on MADNESS

- URLs
  - MADNESS: http://www.csm.ornl.gov/ccsg/html/projects/madness.html
  - MADNESS code repository: http://code.google.com/p/m-a-d-n-e-s-s/
  - SCIDAC project webpage: http://www.csm.ornl.gov/~hqi/scidac/
- Papers (Math)
  - B. Alpert, "A class of bases in L2 for the sparse representation of integral operators," SIAM J. Math. Anal. 24, pp. 246–262, 1993.
  - B. Alpert, G. Beylkin, D. Gines, and L. Vozovoi, "Adaptive solution of Partial Differential Equations in multiwavelet bases," *J. Comp Phys 182*, pp. 149–190, 2002.
  - G. Fann, G. Beylkin, R.J. Harrison, and K.E. Jordan, "Singular operators in multiwavelet bases," *IBM J. Res. & Dev. 48(2)*, pp. 161–171, 2004.

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

54

More on Load Balancing

- Load Balancing
  - J.D. Teresco, K.D. Devine, and J.E. Flaherty, "Partitioning and Dynamic Load Balancing for the Numerical Solution of Partial Differential Equations," in *Numerical Solution of Partial Differential Equations on Parallel Computers,* A.M. Bruaset, P. Bjørstad, eds. Springer-Verlag, 2006.
  - K. Devine, E. Boman, R. Heaphy, B. Hendrickson, and C. Vaughan, "Zoltan data management services for parallel dynamic applications," *Computing in Science and Engineering 4(2)*, pp. 90–97, 2002.
  - Cybenko, G., "Dynamic load balancing for distributed memory multiprocessors," *J. Parallel and Distributed Computing 7,* pp 279–301, 1989.
- Load Balancing in MADNESS
  - R.J. Hartman-Baker, R.J. Harrison, and G.I. Fann, "Load Distribution in MADNESS," *in preparation.*

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

55

Acknowledgments

- Financial support/computing resources
  - Activities performed under contract number DE-AC0500OR22750 between the U.S. Department of Energy and Oak Ridge Associated Universities
  - This research used resources of the Center for Computational Sciences at Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725
- I am grateful for the support of
  - George Fann and Robert Harrison, Computer Science & Mathematics Division, ORNL
  - Mike Heath, University of Illinois at Urbana-Champaign

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

UT-BATTELLE

56