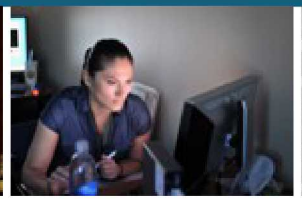
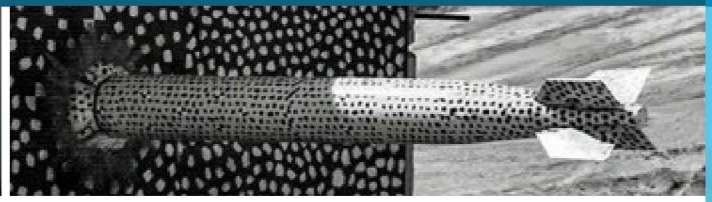
SAND2019-14216C

Sandia National Laboratories

# Task Placement for Reduced Communication Cost





*E3SM All Hands Meeting*

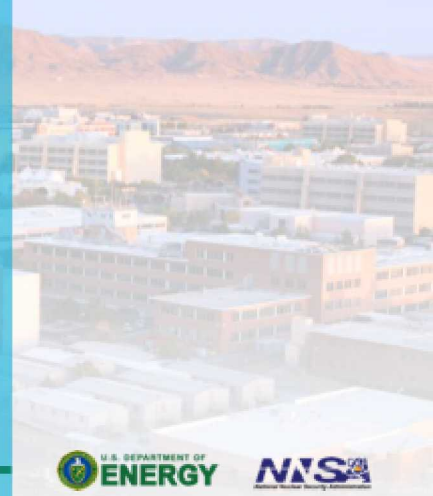
*20 November 2019*

*Presented by*

**J. Austin Ellis** and Karen Devine

Center for Computing Research, Sandia National Laboratories

Albuquerque, NM, U.S.A.



U.S. DEPARTMENT OF ENERGY NNSA

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

# Dragonfly MPI Task Placement Background

**Objective**: Minimize *distance* messages must travel by "mapping" frequently communicating MPI tasks to nearby nodes in allocation.

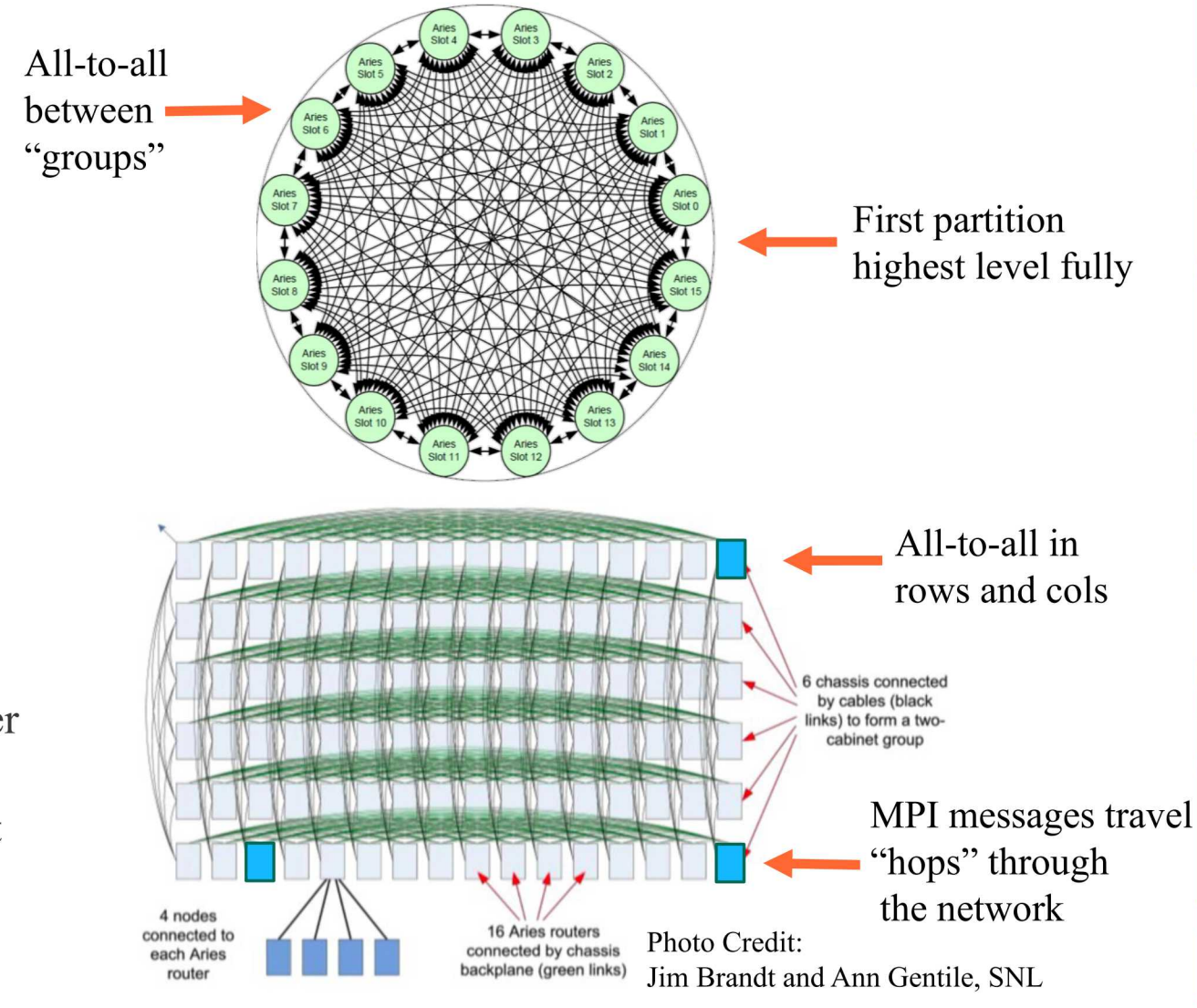
**Contributions:**

New dragonfly task placement algorithm inside Trilinos' Zoltan2 package.

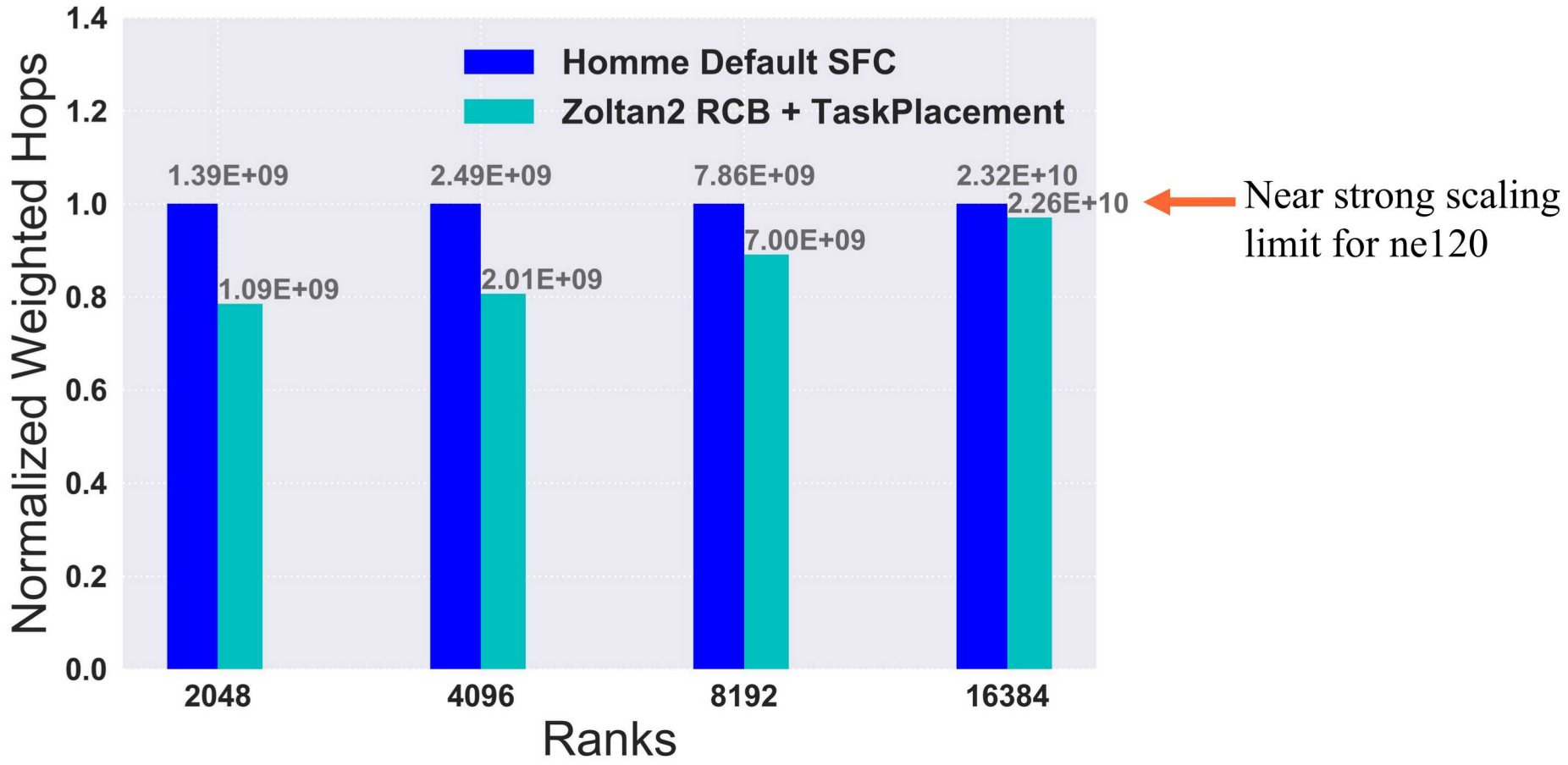
- Use high dimensional coordinate transformation to represent all-to-all connections
- Use coordinate stretching to match bandwidths and common congestion.

Added new capabilities to Zoltan2's Multi-Jagged (MJ) geometric coordinate partitioner.

- Will compute a nonuniform partitioning based on a user provided distribution
- Ensures the "group" dimension is fully partitioned first

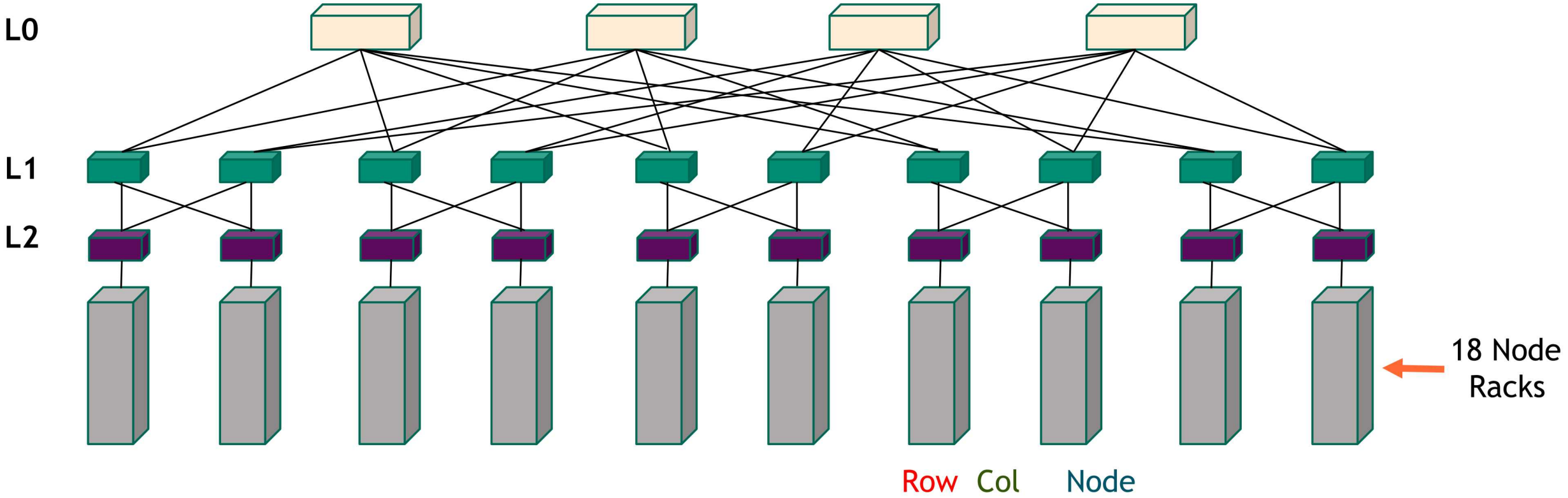


# Preliminary Numerical Results from Theta




- **Test Case**: ALCF's Theta machine, HOMME v1 ne120 benchmark, 16 ranks per node
- Geometric task placement increases task locality in the network, decreasing distances messages must travel
- Weighted hops decreases by **up to 20%**
- Observed decreased runtime variability (50% on Cori, harder to quantify on Theta)

# Future Work: Fat Tree Task Placement on Summit




- Obtain Summit node floor location using `gethostname()`: **[A-H][01-36]n[01-18]**
- Transform floor coordinates to network **switch** "neighborhoods"
  - Ex. All nodes in **A01-A18** connected in a 3-hop neighborhood, **A19-A36** are a separate 3-hop neighborhood
- Use recursive 3-level nonuniform MJ partitioning (L0 → L1 → L2)
- Targeting full system scale runs for HOMME

# Thank you!