

SUPPORT FOR LOAD BALANCING ON BLUE WATERS

Principal Investigator

Laxmikant Kale (kale@illinois.edu)

Team

Ronak Buch (rabuch2@illinois.edu)

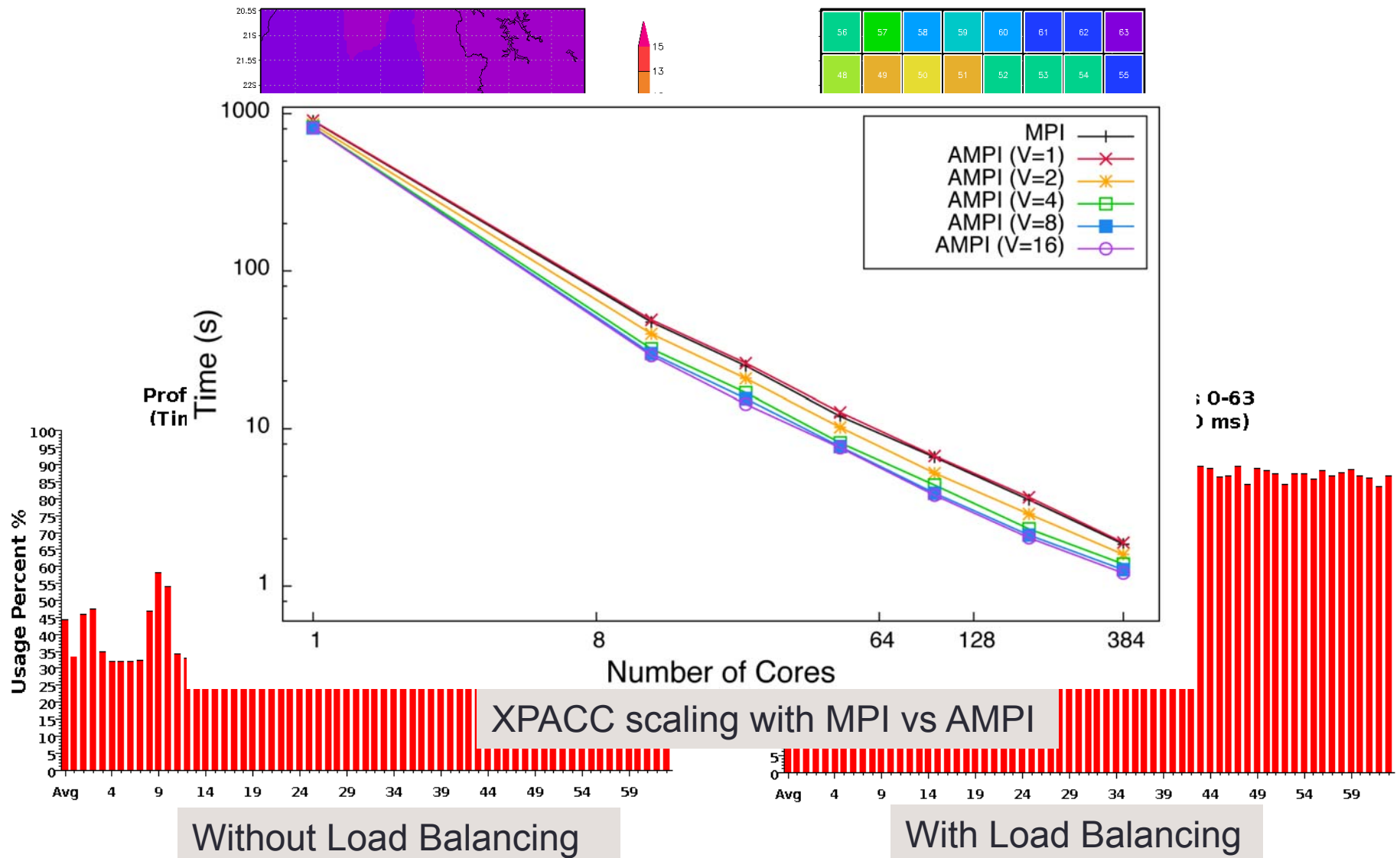
Harshitha Menon (gplkrsh2@illinois.edu)

Nikhil Jain (nikhil@illinois.edu)

Load balancing

- Load imbalance can be a critical factor that hurts performance
- Heterogeneous resources presents a new challenge for balancing load
- Charm++ load balancing framework
 - Measurement based load balancing framework
 - Based on the principle of persistence or user provided metric
 - Various load balancing algorithms built in
 - Fully automated : migration, when to call LB
- AMPI: MPI implementation on top of Charm++ to enable features such as load balancing and fault tolerance in MPI programs

Load balancing



Planned Work

- Heterogeneous load balancer
 - Balance work between GPUs and CPUs
 - Heterogeneous task delegation with different scheduling strategies
- Generic library with a suite of load balancing algorithms
 - MPI applications with migratable entities that can provide information about work units via API can use this library to get load balancing decisions
- Provide support to port MPI applications to AMPI (MPI implementation on top of Charm)

Collaboration Objective

- Assess applications and identify load imbalance problem
- Help the application teams with heterogeneous load balancing problem
- Provide support for MPI applications with migratable entities to use the load balancing library
- Identify MPI applications that can benefit from Charm++ model and help convert them to AMPI