PARALLEL PROGRAMMING WITH MIGRATABLE OBJECTS: CHARM++ IN PRACTICE

Presenter: Harshitha Menon

PPL Group
Abhishek, Akhil, Bilge, Ehsan, Eric, Lukasz, Michael, Nikhil, Ronak,
Yanhua, Xiang, Laxmikant Kale

INTRODUCTION

RTS DESIGN

MINI-APPLICATION

INTRODUCTION

Application-Driven Variations

RTS DESIGN

DESIGN ATTRIBUTES

Over-decomposition

Migratability

Asynchronous Message-Driven Execution

OVER-DECOMPOSITION

Decompose work & data units to many more pieces than execution units

MIGRATABILITY

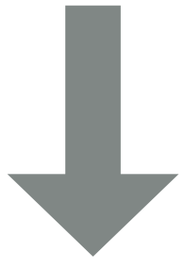
Move work units to another execution unit at run time.

ASYNCHRONOUS MESSAGE-DRIVEN

Work units are scheduled when the message arrives.

ADAPTIVE & POWERFUL RTS

Over-decomposition + Migratability + Asynchrony



Adaptive RTS

RTS FEATURES

LOAD BALANCING

FAULT TOLERANCE

POWER AWARENESS

MALLEABILITY

COMM-OPTIMIZATIONS

CONTROL SYSTEM

INTEROPERATION

...

LOAD BALANCING

Load imbalance is a critical factor that affects performance

Over-decomposition with migratability enables LB

Charm++ load balancing framework

FAULT TOLERANCE

State of the application is checkpoint to disk or memory

RTS automatically detects faults and restart the work units

Charm++ provides various schemes

POWER AWARENESS

One way to save energy - cooling energy

RTS controls the temperature using DVFS

RTS triggers load balancing when required

MALLEABILITY

Ability to shrink and expand jobs

Improve the cluster utilization

RTS automatically handles this

COMM-OPTIMIZATION USING TRAM

Fine-grained messages can create a lot of overhead

TRAM aggregates fine-grained messages into larger messages

INTROSPECTIVE CONTROL SYSTEM

Can handle dynamicity without burdening the programmer

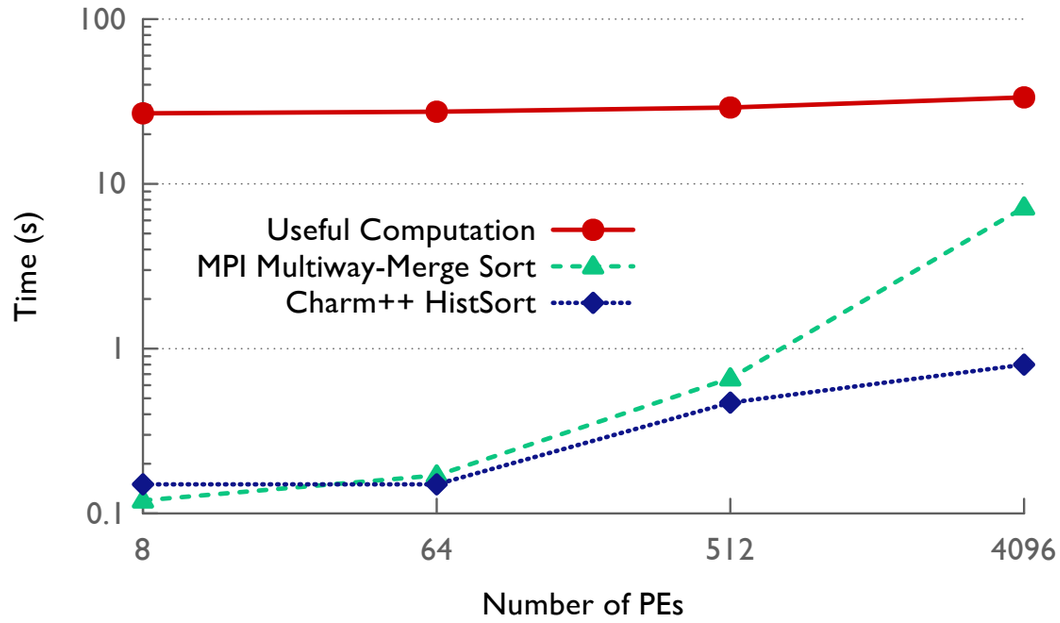
Monitors the application and performs analysis

Reconfigure the application

INTEROPERATION

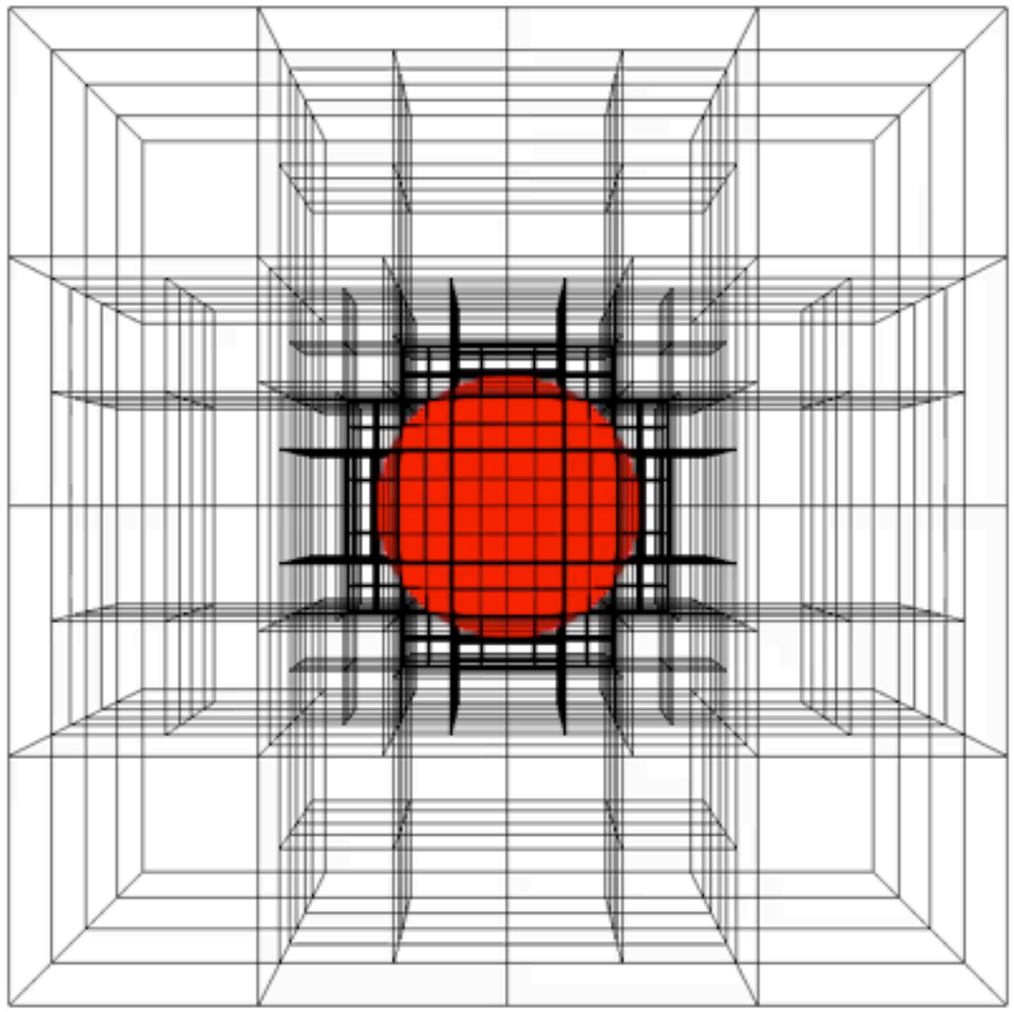
-Modules implemented in MPI and Charm++ can interoperate

-Used it CHARM to use the parallel sort library in Charm

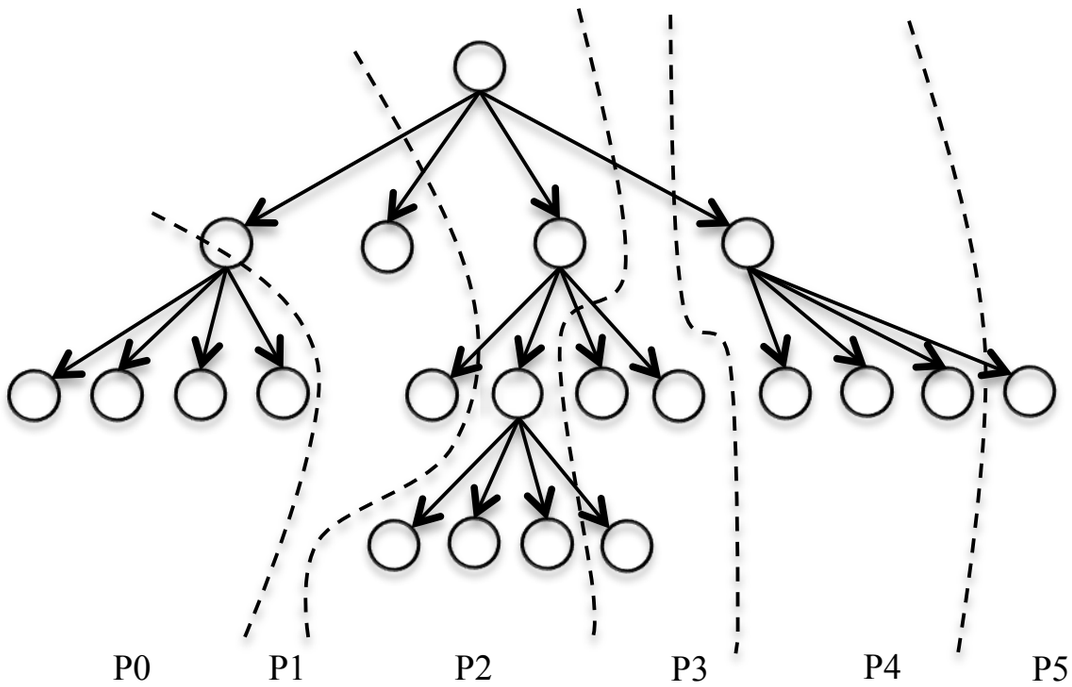


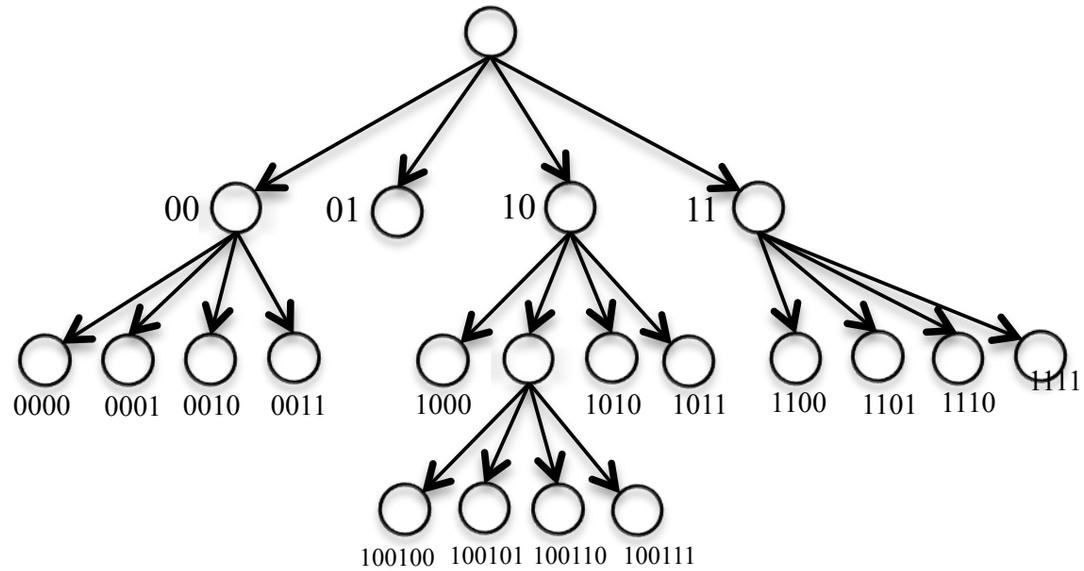
MINI-APPS

ADAPTIVE MESH REFINEMENT (AMR)



AMR3D





Process based

-Contiguous blocks assigned to a process

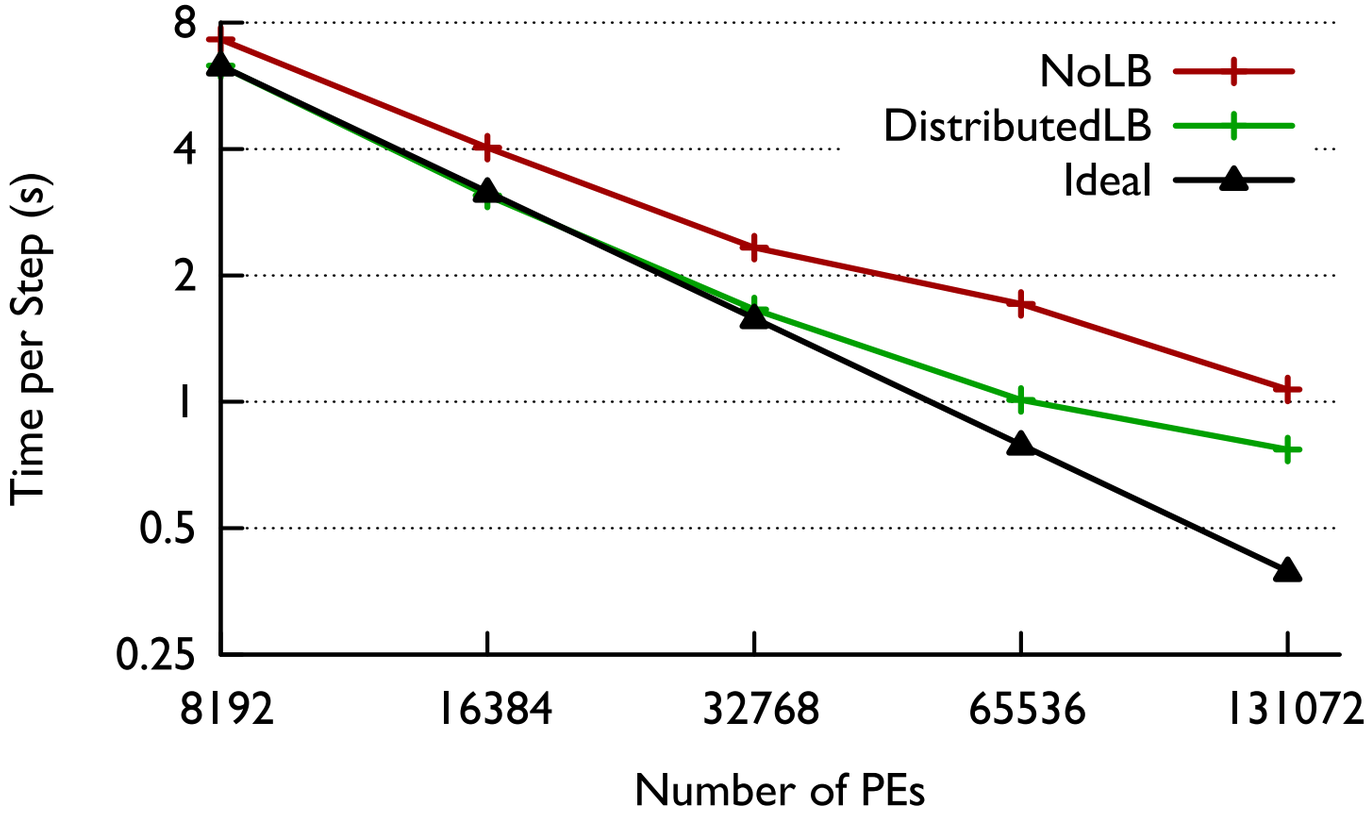
Object based

-Each block is an independent migratable object

-During refinement new blocks are created

-During coarsening blocks are deleted

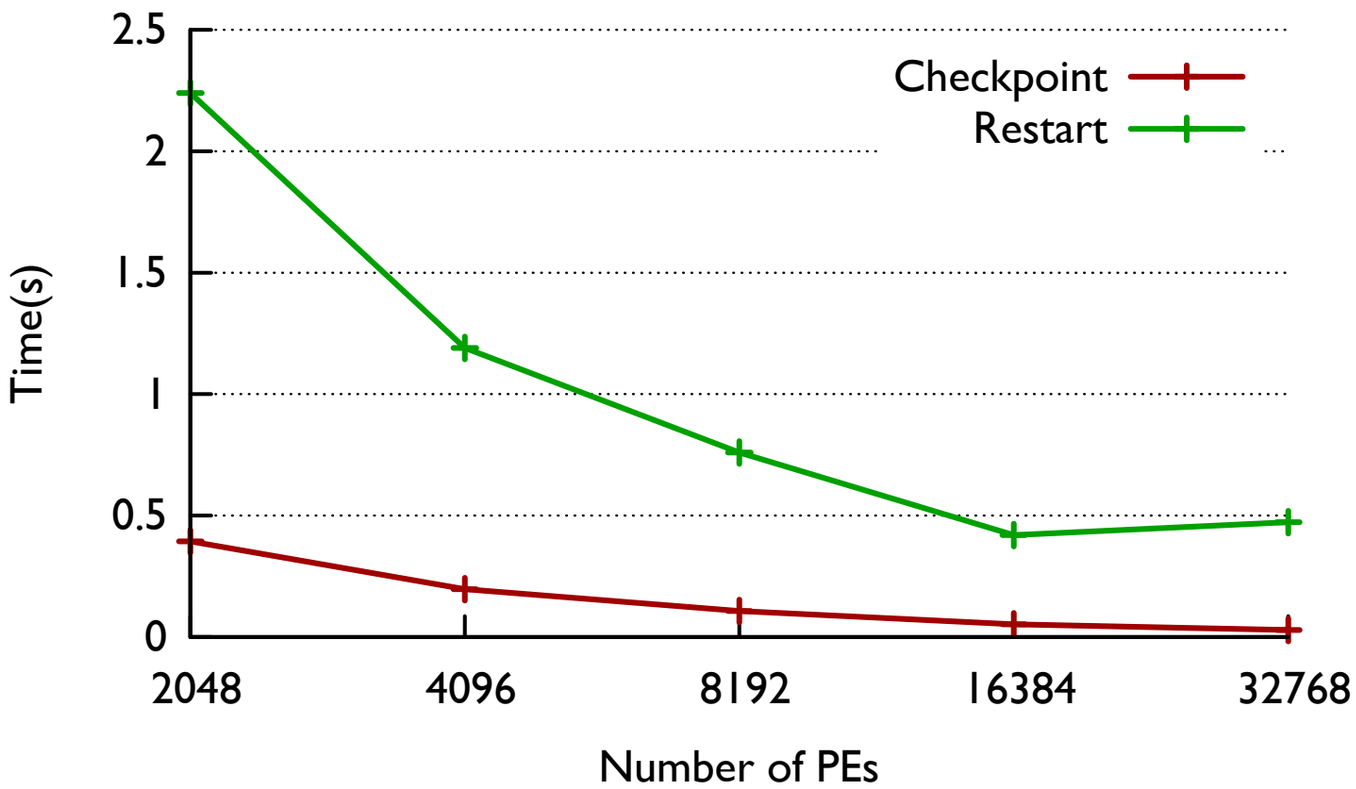
AMR3D



-IBM BG/Q on up to 128K cores

-DistributedLB gives 40% benefit at 128K cores

AMR3D

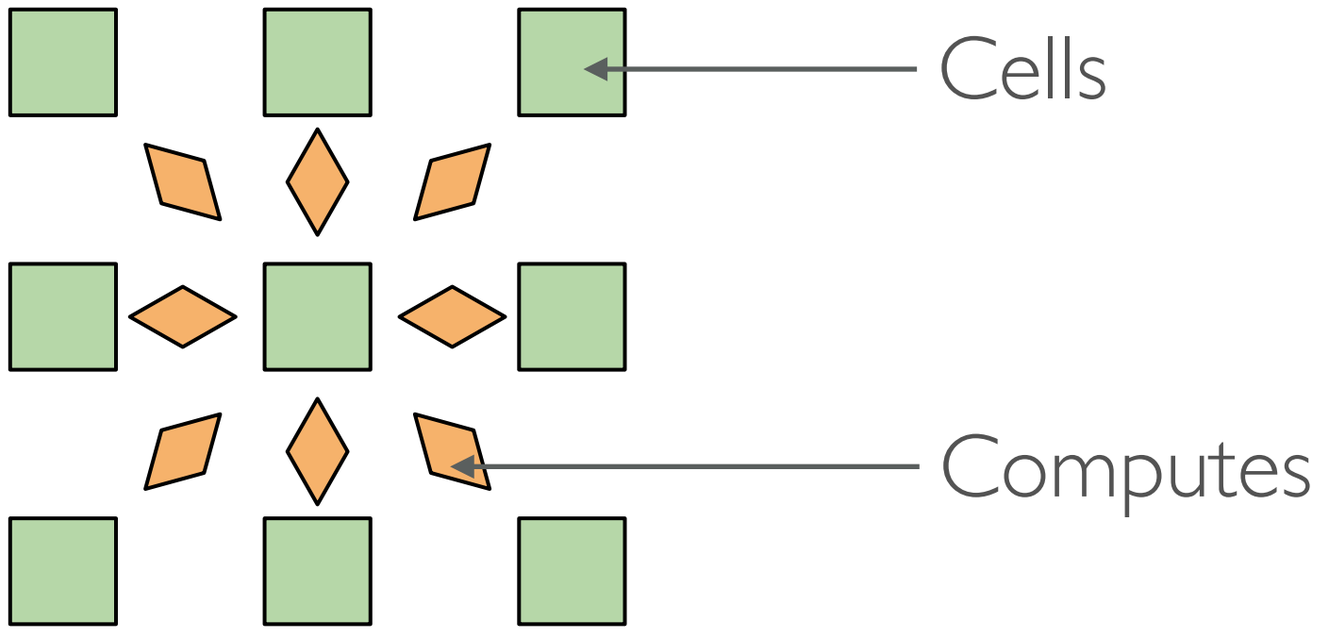


-In-memory checkpoint

-Checkpoint and restart time decreases

-At 32K cores, checkpoint time 29ms and restart time is 470ms

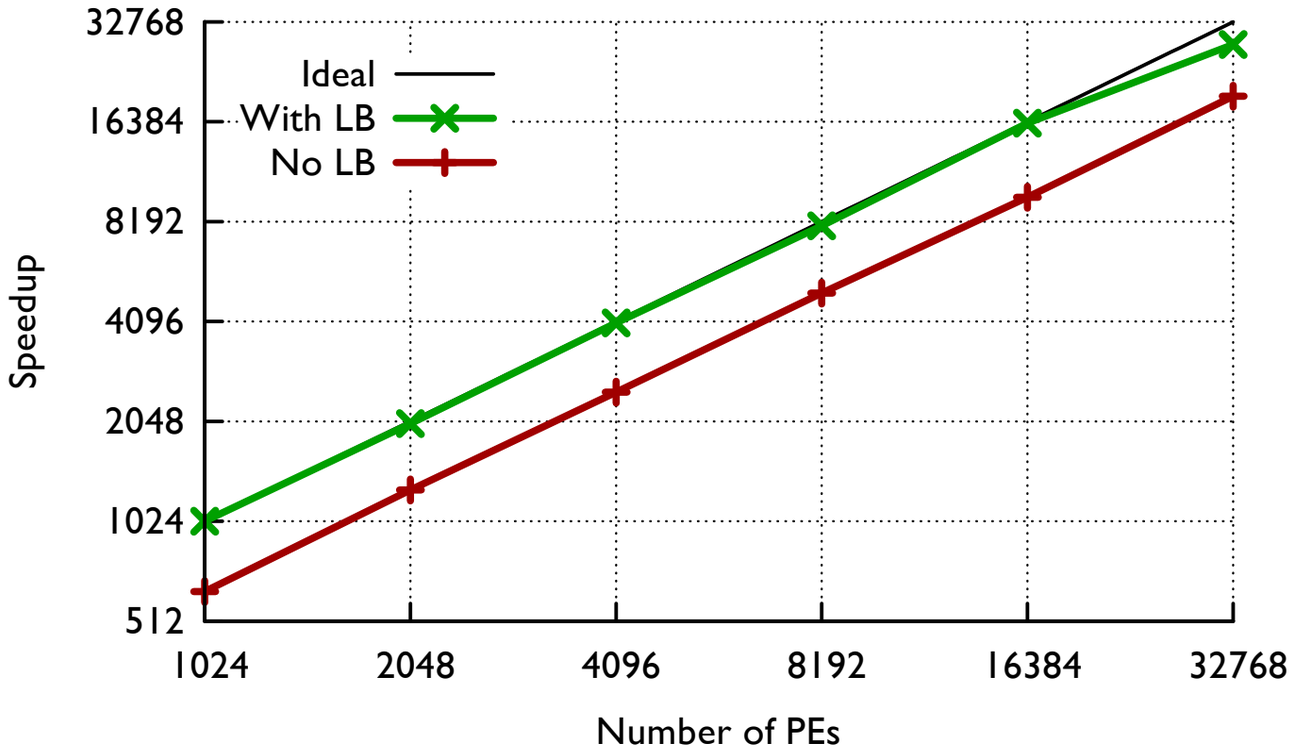
MOLECULAR DYNAMICS - LEANMD



Cells: 3D chare array representing 3D decomposition of atoms

Computes: Sparse 6D chare array which performs force calculations

LEANMD

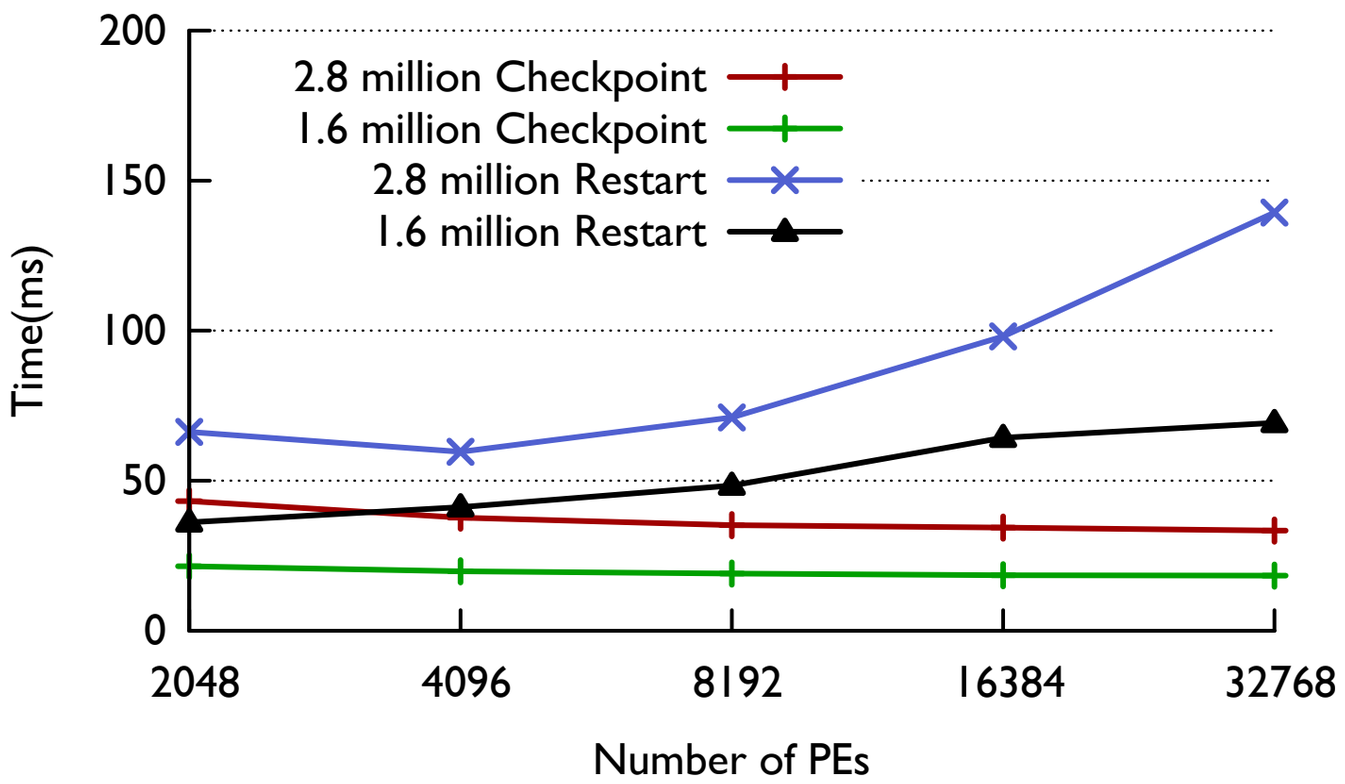


-2.8 million atoms system

-IBM BG/Q on up to 32K cores

-Dynamic adaptive load balancing with Hierarchical LB

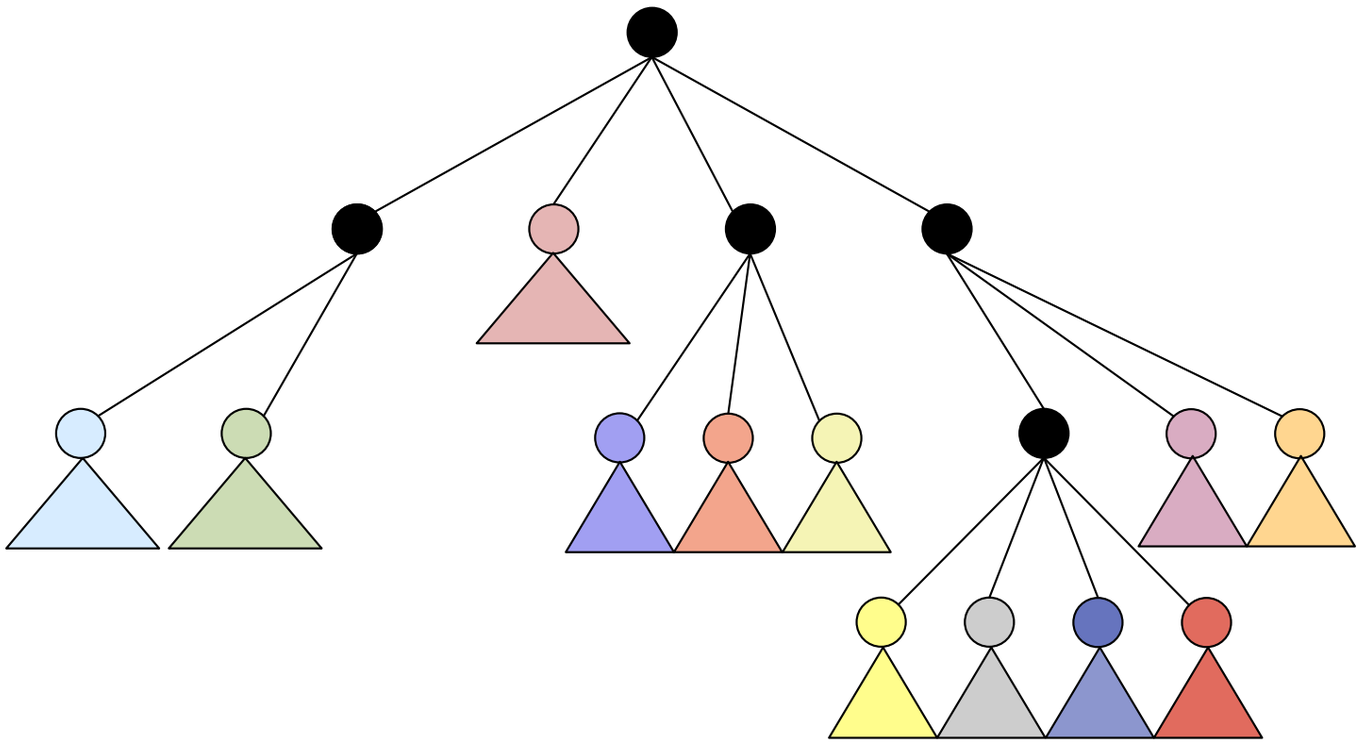
LEANMD



-In-memory checkpoint restart with simulated failures

-At 32K cores, checkpoint time 33ms and restart time is 139ms

BARNES HUT



Decomposition

3D decomposition of the space into chare array called *TreePiece*

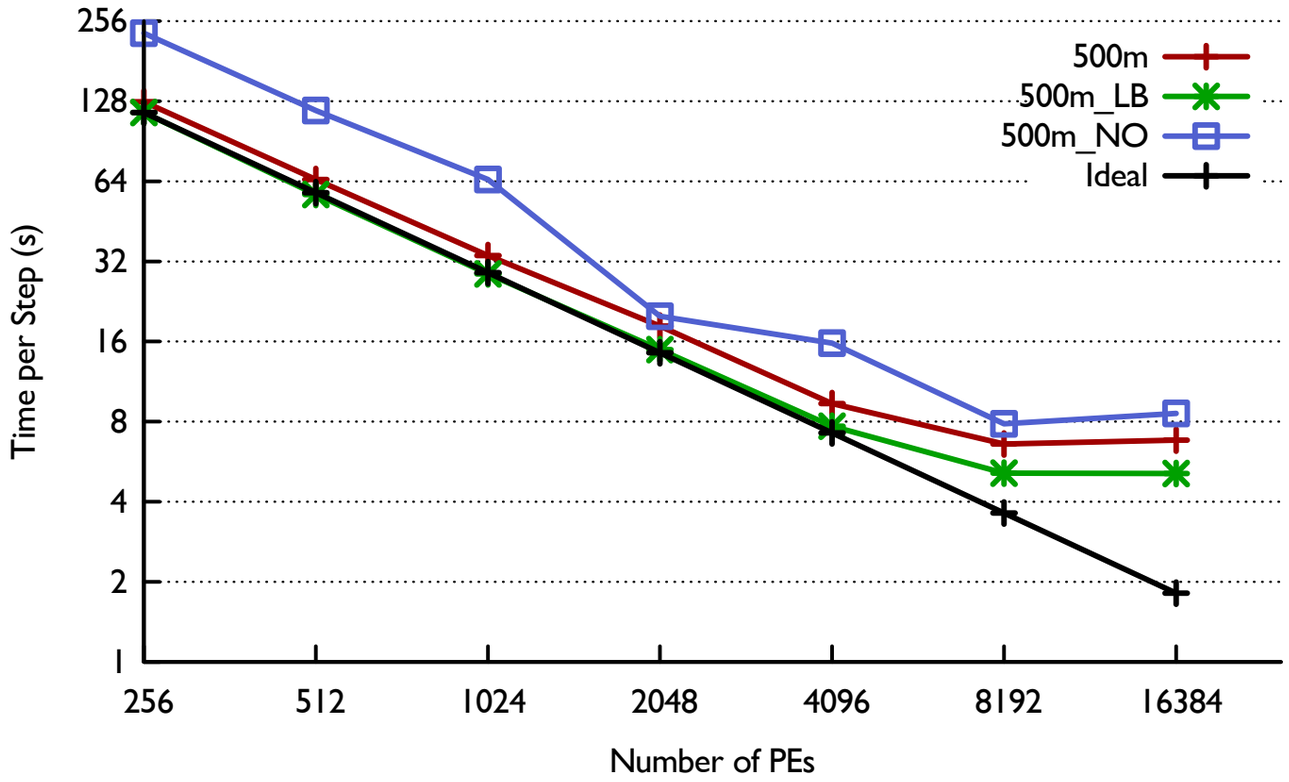
Prioritized Messages

Remote work higher priority than local work

Load Balancing

Specialized OrbLB

BARNES HUT



-500 million particles system

-Blue Waters on up to 16K cores

-Over-decomposition and LB improves performance

PARALLEL DISCRETE EVENT SIMULATION (PDES)

Decomposition

Logical Processes (LP) execute discrete events

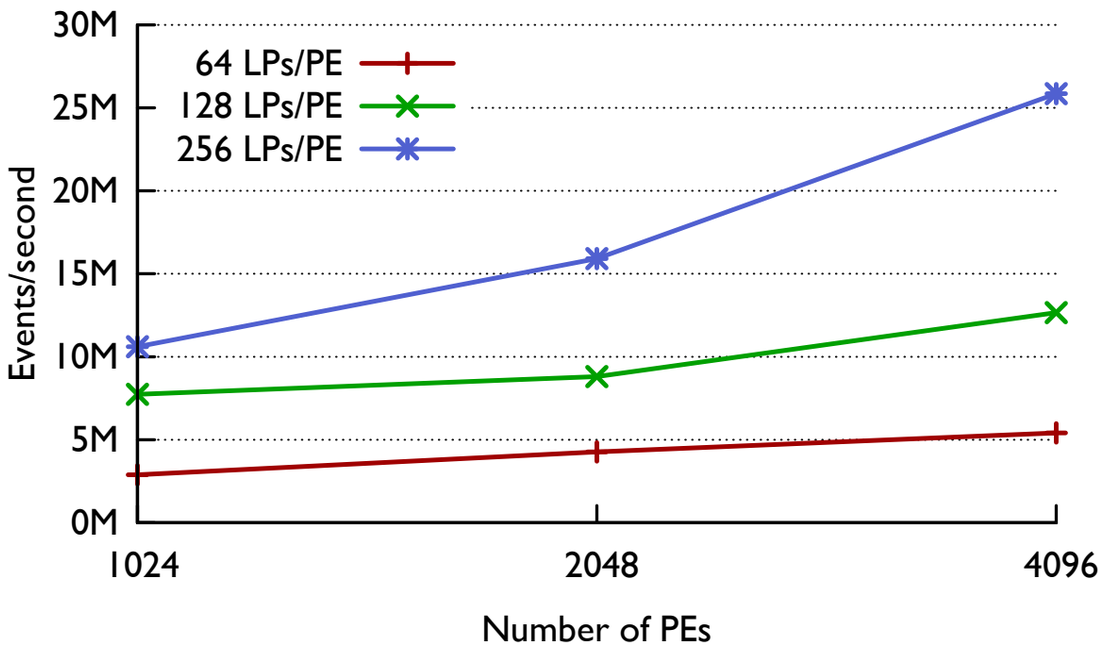
Asynchronous Message Driven

Communication pattern cannot be determined *a priori*

TRAM

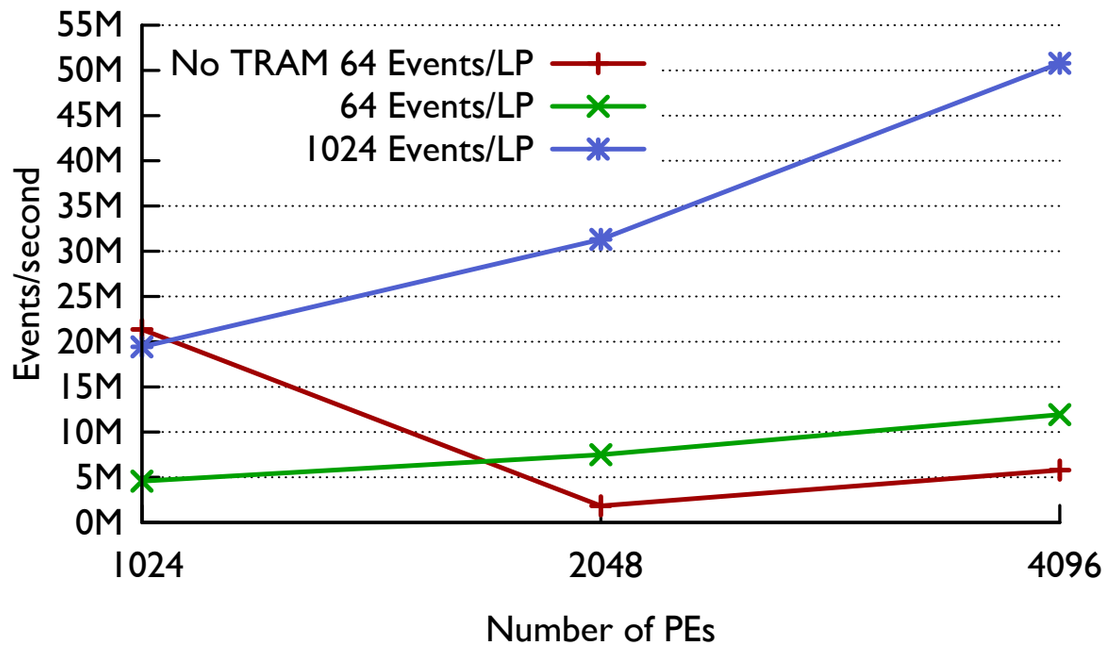
Fine-grained messages optimized using TRAM

PDES



-PHOLD simulation benchmark

-Overdecomposition increases the event rate



-At high communication volume using TRAM improves the event rate.

LULESH - AMPI APP

AMPI

MPI app ported to AMPI with minimal effort

Can use Charm++ features

Over-decomposition

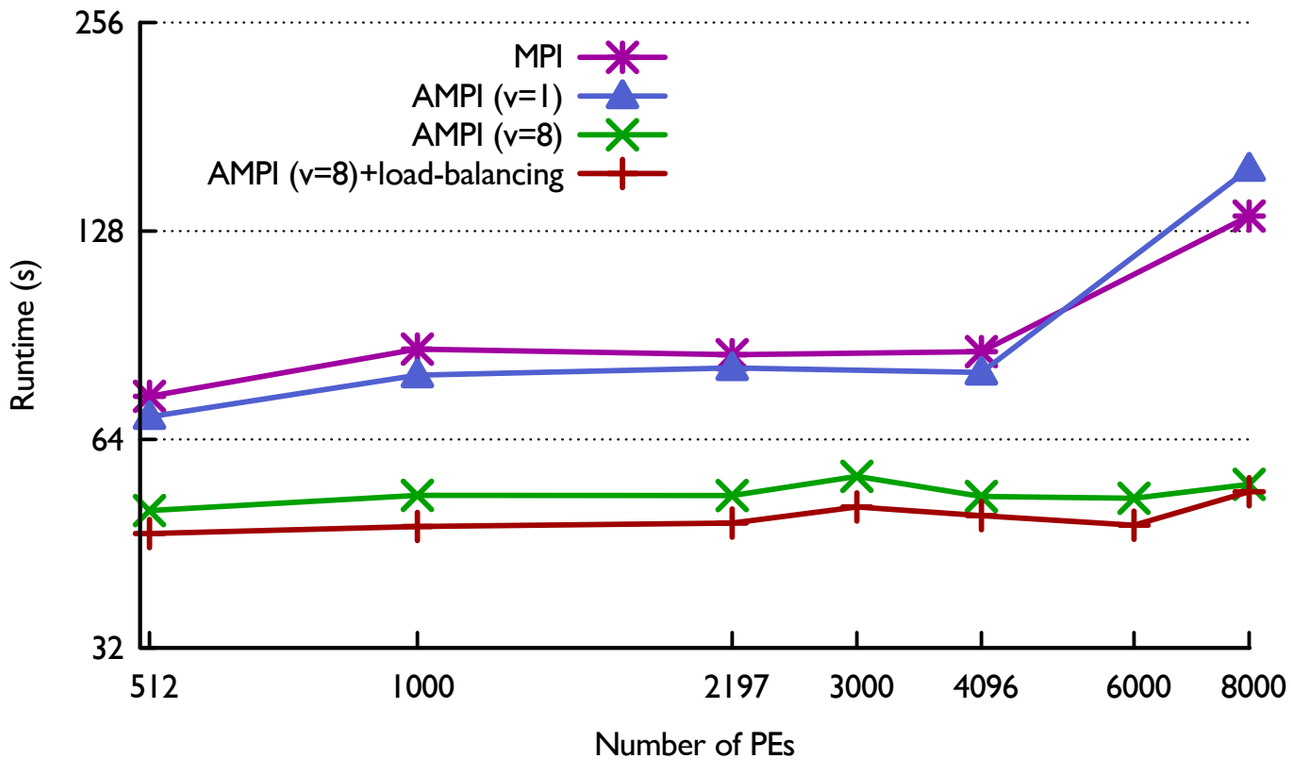
Gives cache benefits

Dynamic Load Balancing

Any of the Charm++ load balancers can be used

Any number of cores (Not necessarily cubic)

LULESH



-AMPI run with virtualization gives speedup of 2.4 over MPI and without virtualization

-Automatic load balancing is able to handle the small amount of imbalance

-AMPI version can run on non-cubic number of cores

SUMMARY

Shown mini-apps which use the Charm++ design model and features to scale efficiently

http://ppl.cs.illinois.edu/papers

THANK YOU!

AMR

Decomposition

Each block is an independent migratable object

During refinement new blocks are created

During coarsening blocks are dynamically deleted

Load Balancing

Distributed dynamic load balancing

Fault Tolerance

In-memory checkpoint restart

MOLECULAR DYNAMICS - LEANMD

Decomposition

Cells: 3D chare array representing 3D decomposition of atoms

Computes: Sparse 6D chare array which performs force calculations

Load Balancing

Dynamic *hierarchical* load balancing with *MetaBalancer*

Fault Tolerance

In-memory checkpoint restart

RTS FEATURES

LOAD BALANCING

FAULT TOLERANCE

POWER AWARENESS

MALLEABILITY

COMM-OPTIMIZATIONS

CONTROL SYSTEM

INTEROPERATION

...

OVER-DECOMPOSITION