What is NERSC and How Can Their Computer Resources Help You?

Richard Gerber, NERSC

Noel Keen, CRD/EETD

Wangda Zuo, EETD

Motivation: A Powerful Tool, Not a NERSC Promotion

- I've been working part time with the Building Simulations Group in EETD and the idea of a general introduction to parallel programming was discussed.

- Recruited Richard Gerber at NERSC to help. Richard has been a High Performance consultant at NERSC for many years and has a PhD in computational physics.

- Wanted to add Wangda Zuo to give an example of using parallel programming with a GPU.

- I certainly do not claim to know the computing maturity of projects in EETD as a whole. These topics may be things everyone already knows.

- I've been an active user of HPC centers (mostly NERSC) and it feels like the capabilities of the machines are growing so rapidly, that maybe we should change the way we work to take full advantage. Parallel computing and HPC may not be necessary for good science/engineering – it's only a tool.

Parallel Programming Models

|	Hardware Affected	Maturity	General Programming Difficulty	Availability/Portability	Current Max Level of Parallelism
Threading (OpenMP, PThreads, TBB)	Processor	Good	Easy to medium. Allows incremental parallelism	Everywhere (TBB is Intel only)	50
GPU (OpenCL, CUDA)	GPU/Processor	Very New	Difficult, esp to achieve near max performance	Modern Graphics Cards, Specs changing fast (CUDA is Nvidia only)	1000
MPI	Nodes/Network	High	Medium to Difficult	Everywhere (easy to install)	Millions
UPC, CoArray Fortran, PGAS	Nodes/Network	New	Difficult	Depends. Not as easy as MPI.	Millions
Multiple Serial Jobs (Cloud?)	Nodes/Network	High	Easiest	Everywhere	Millions

Two Examples of Using NERSC in EETD Projects

○ A project in the Airflow and Pollutant Transport Group was using many hours of serial computing to perform numerical optimizations using E+ results. I helped to implement a parallel version of the optimization (Particle Swarm Optimization) and then to run on NERSC machines. Routinely ran 128-way jobs for 12 hours at a time.

○ I've been working in the Building Simulations Group to improve the runtime performance of E+. Recently I've tried using OpenMP to thread 2 major loops. I was able to see a speedup of about 2-3x for some problems and no speedup with others. I used NERSC primarily to submit several concurrent jobs for timing and OpenMP parameter searching. (10 jobs @ serial, 1,2,3,4,6,8 threads = 70 runs)

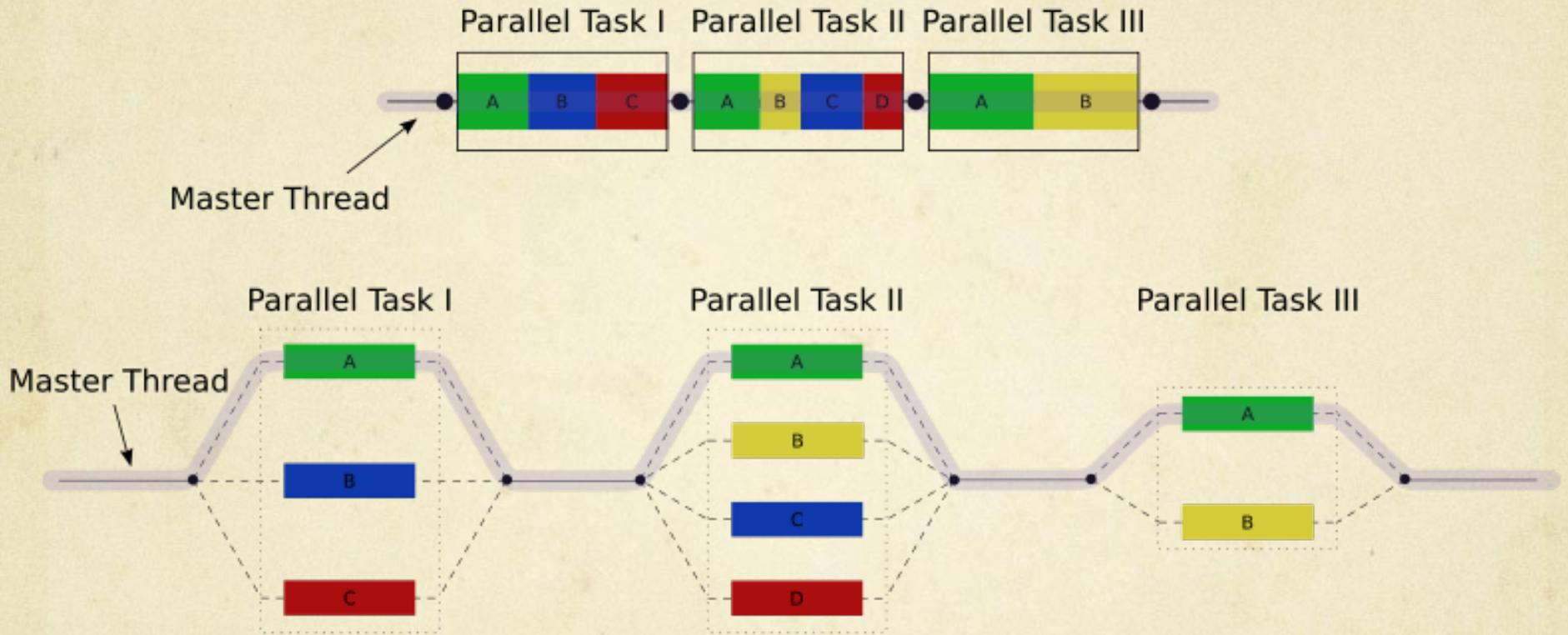# Particle Visualization using Parallel Coordinates
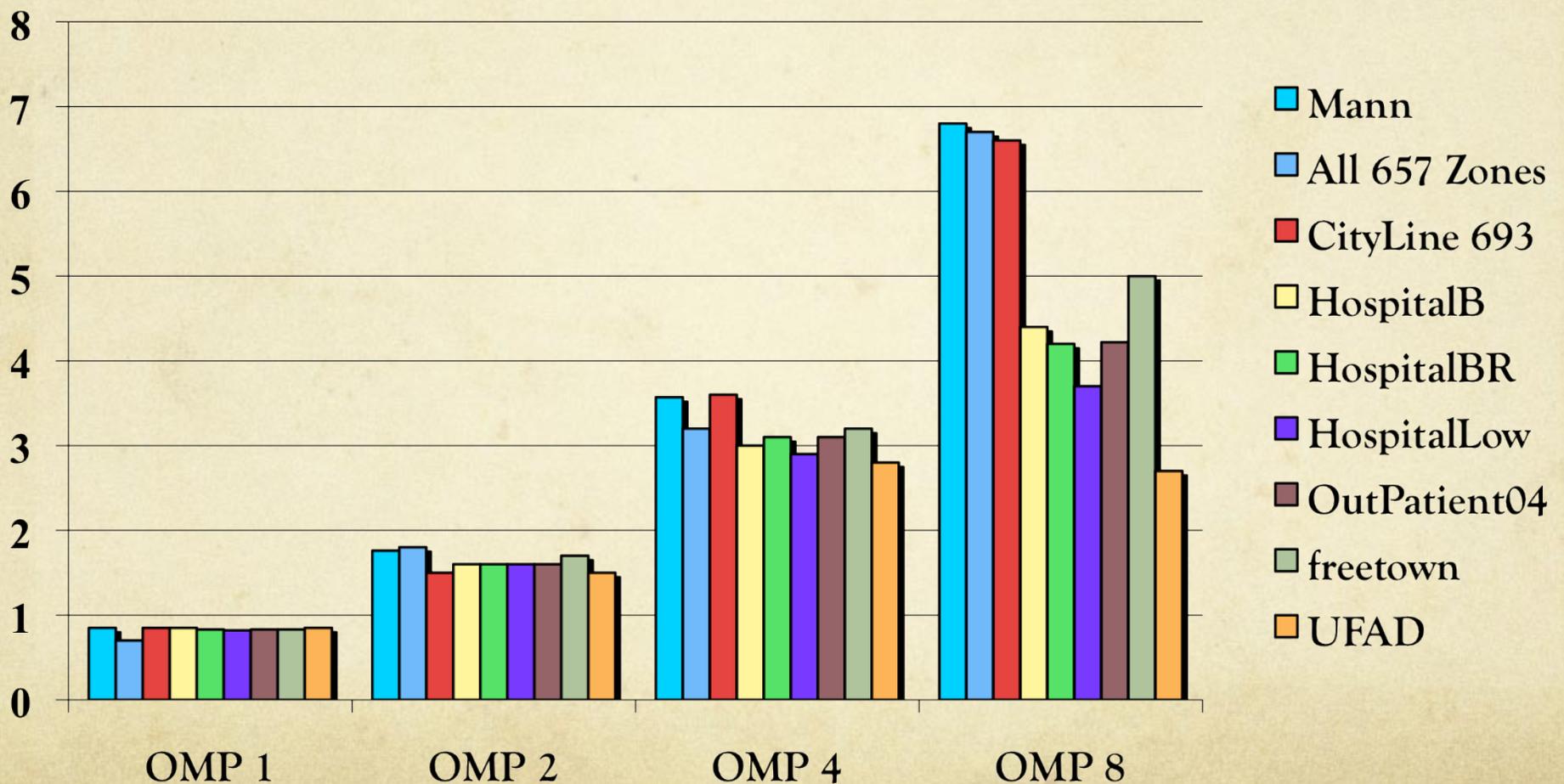
(courtesy of Viz group in CRD)
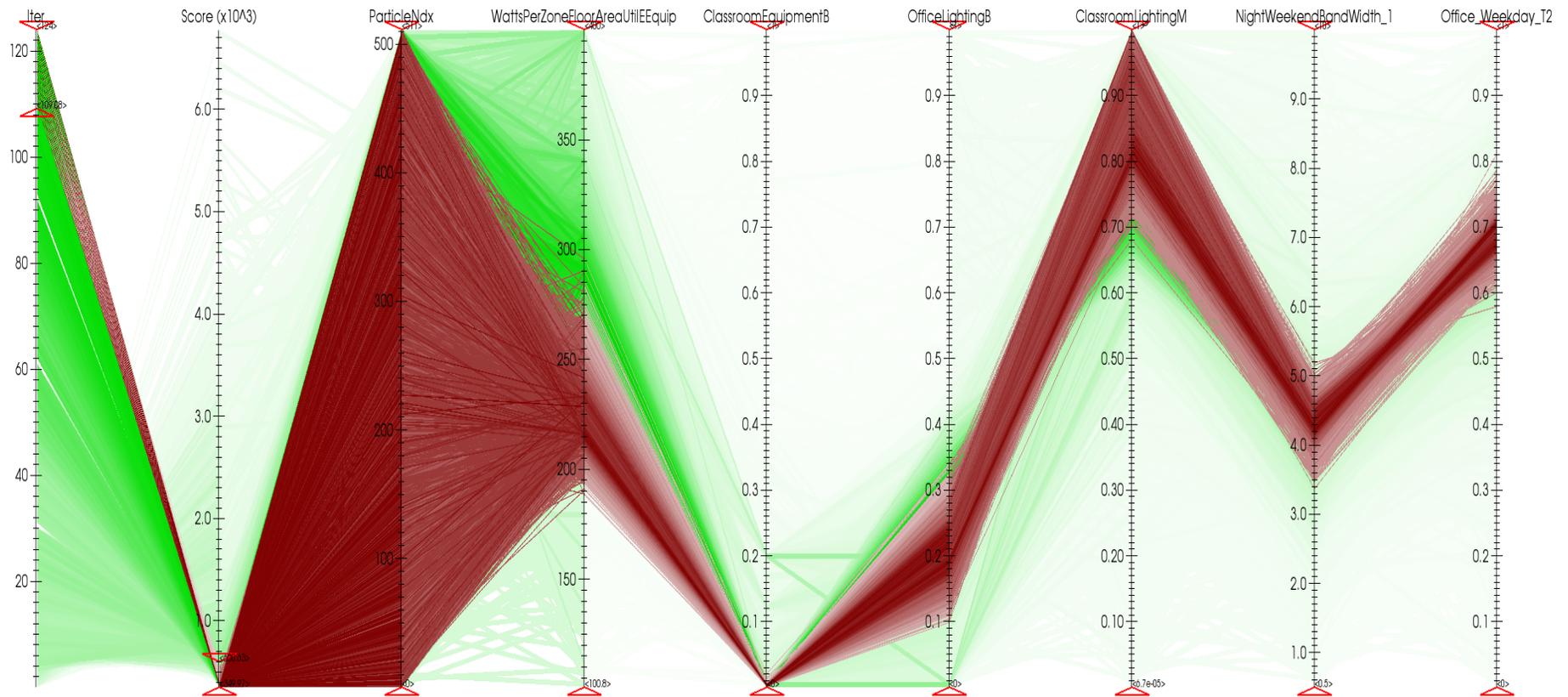
Percent Time in 2 Major Loops of E+ targeted for Thread Parallelism
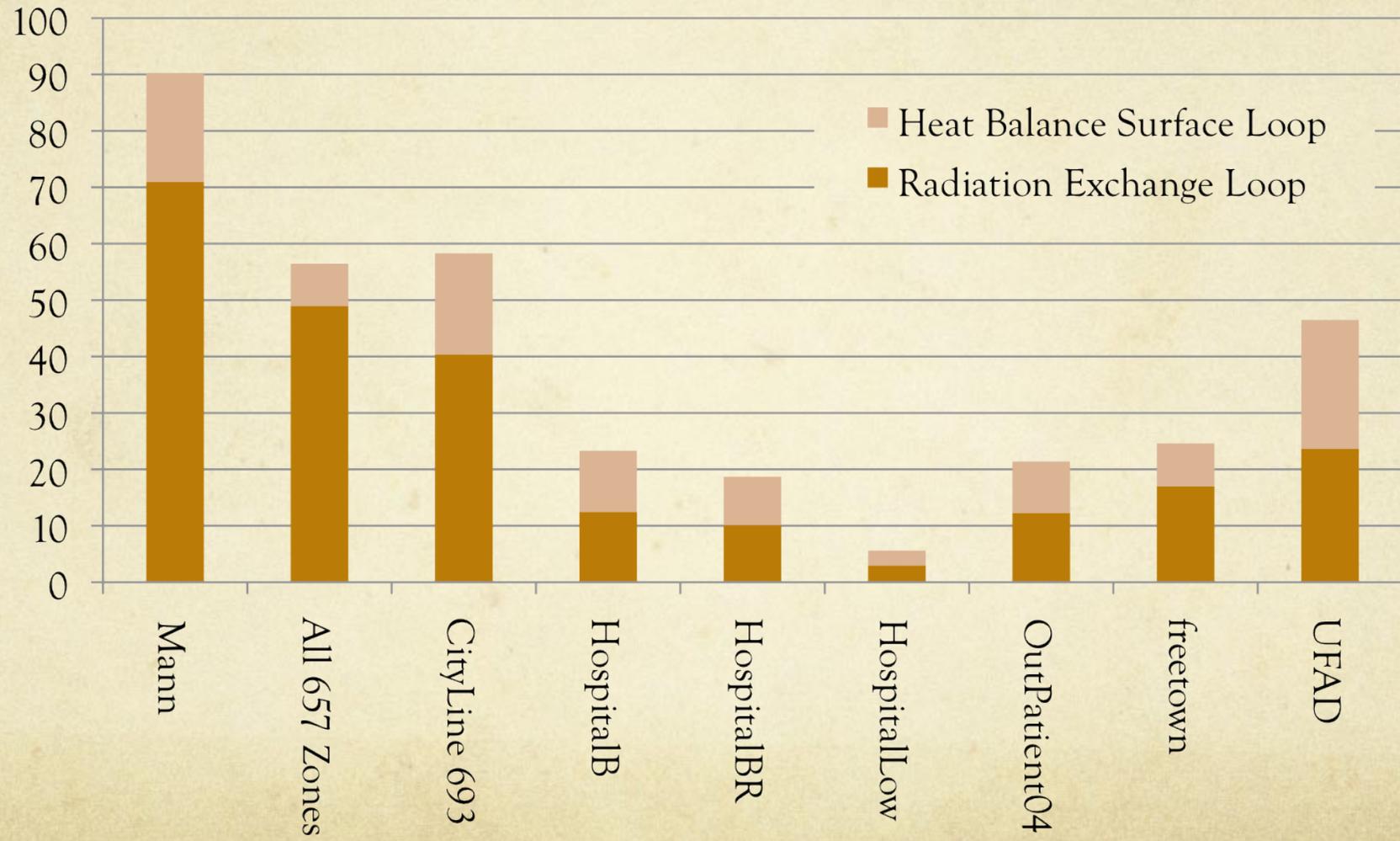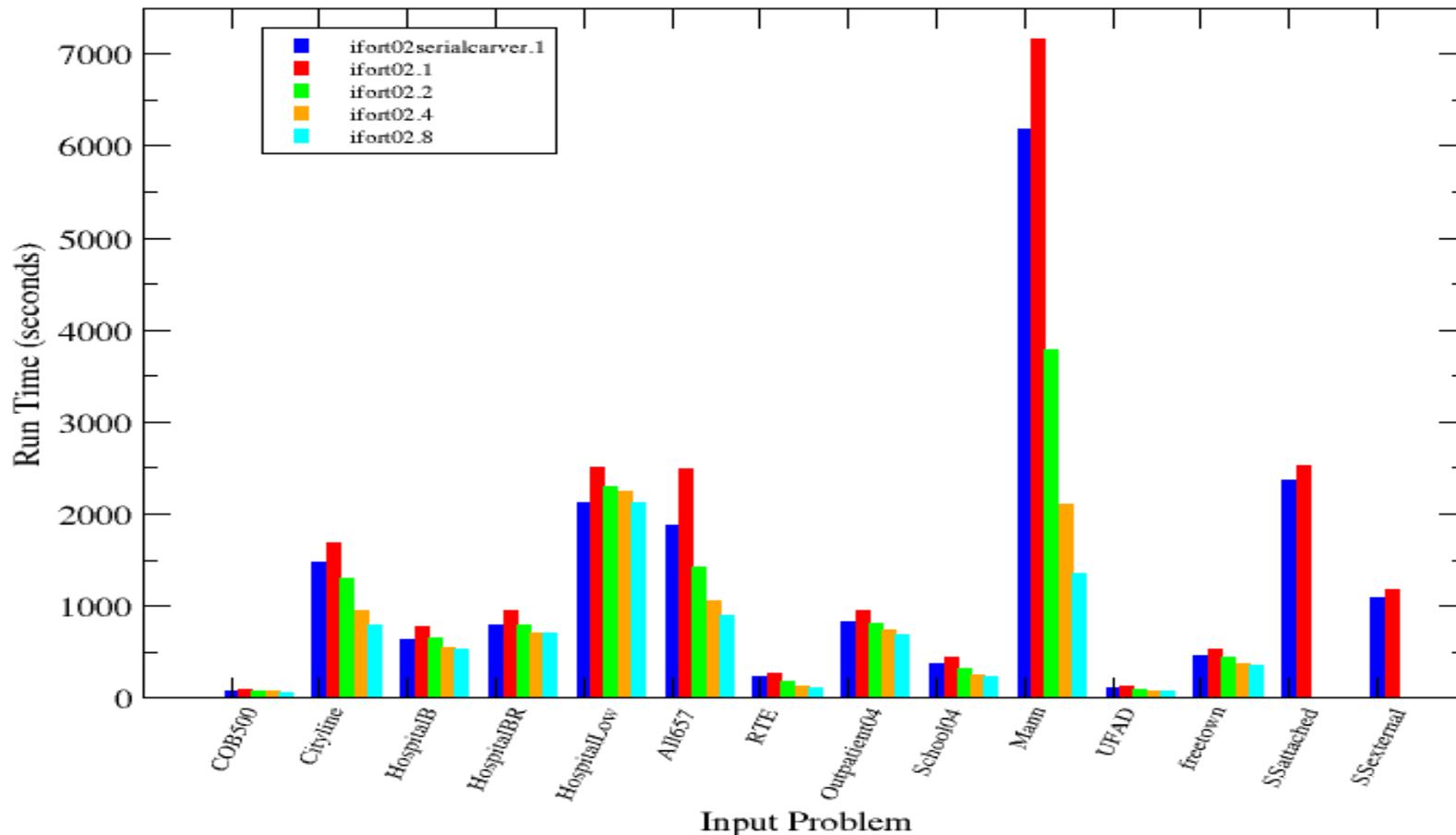
Basic Concept of Threading

Speedup in Radiation Exchange Loop with Threads vs Serial

Runtime Improvements to E+ using OpenMP on Carver at NERSC

Run Time of Performance Benchmarks
Carver, Intel V11.1

Runtime Improvements to E+ with OpenMP (Speedups)

Run Time of Performance Benchmarks
Carver, Intel V11.1