Parallelization of the Z-Model

Thomas Hines

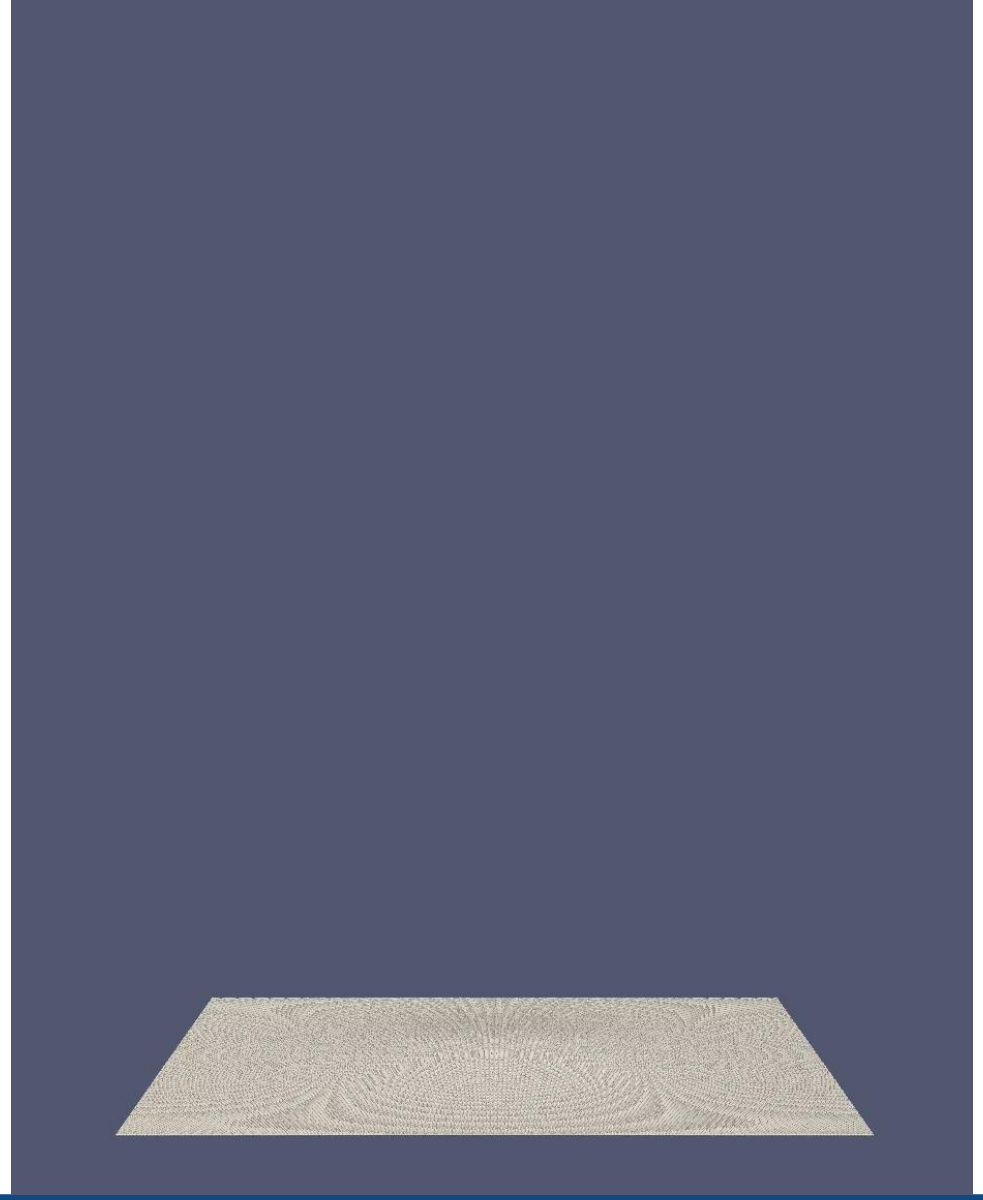
University of Tennessee Chattanooga

8/23/2021

What is Z-Model?

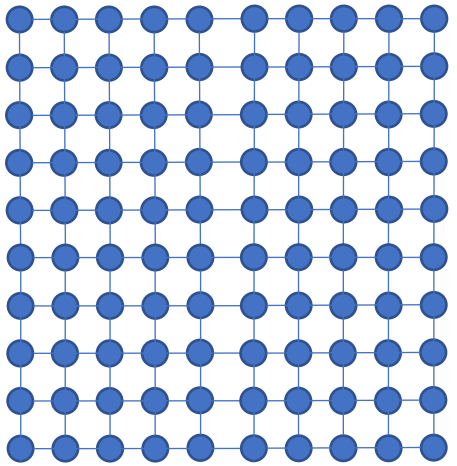
- Simulates the interface between two fluids
- Rayleigh-Taylor Instability or Richtmyer-Meshkov Instability
- By Steve Shkoller at UC Davis

- Raaghav Ramani and Steve Shkoller. “A multiscale model for Rayleigh-Taylor and Richtmyer-Meshkov instabilities”. In: Journal of Computational Physics 405 (2020), p. 109-177.



What is Z-Model?

- 2D mesh of points
- Two types of interactions
  - Neighbor interactions along the mesh
  - Global interactions between all points



What is Z-Model?

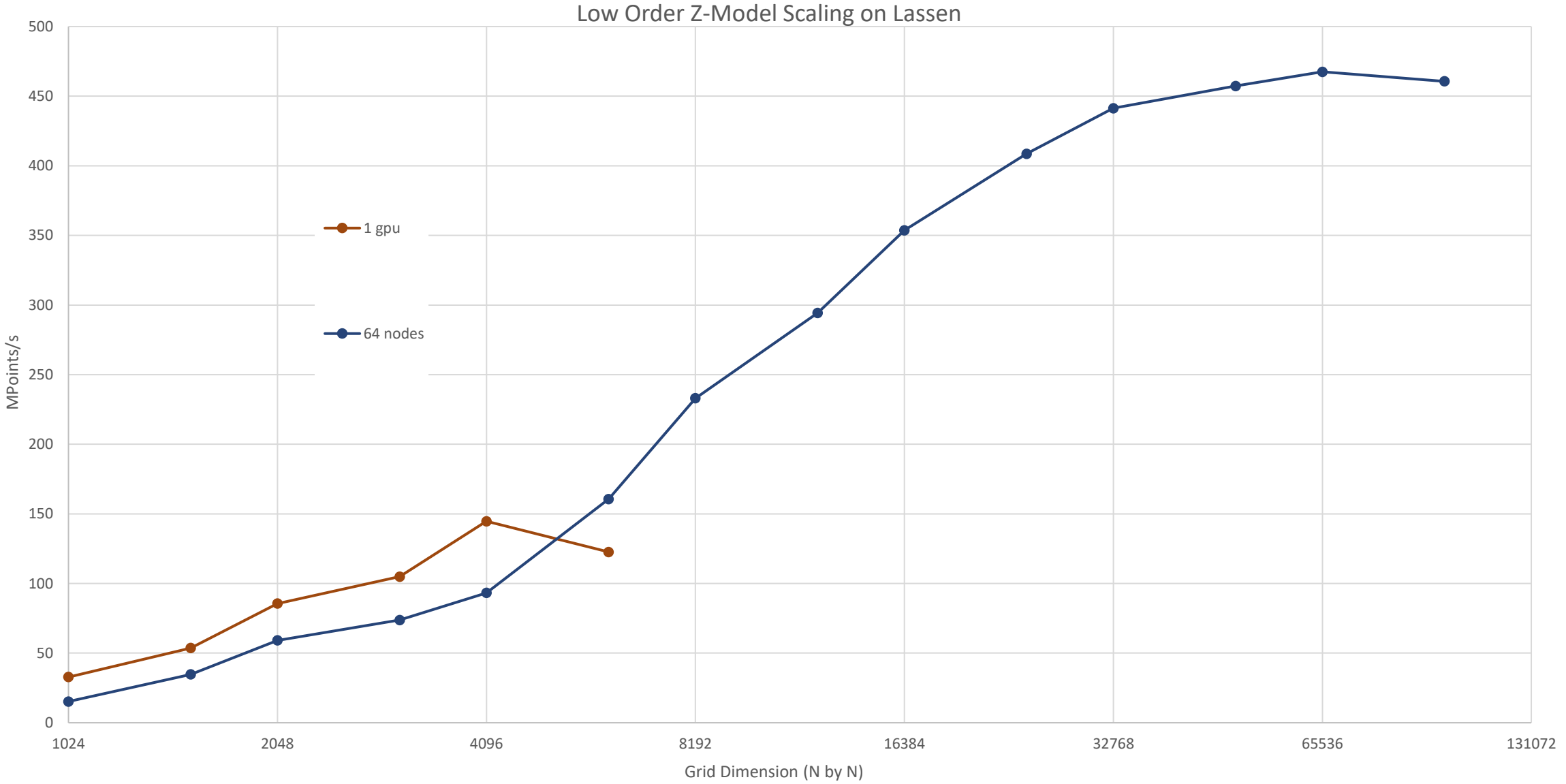
- Two Z-Models
  - Low Order - FFTs to approximate global forces
  - High Order - Direct calculation of global forces

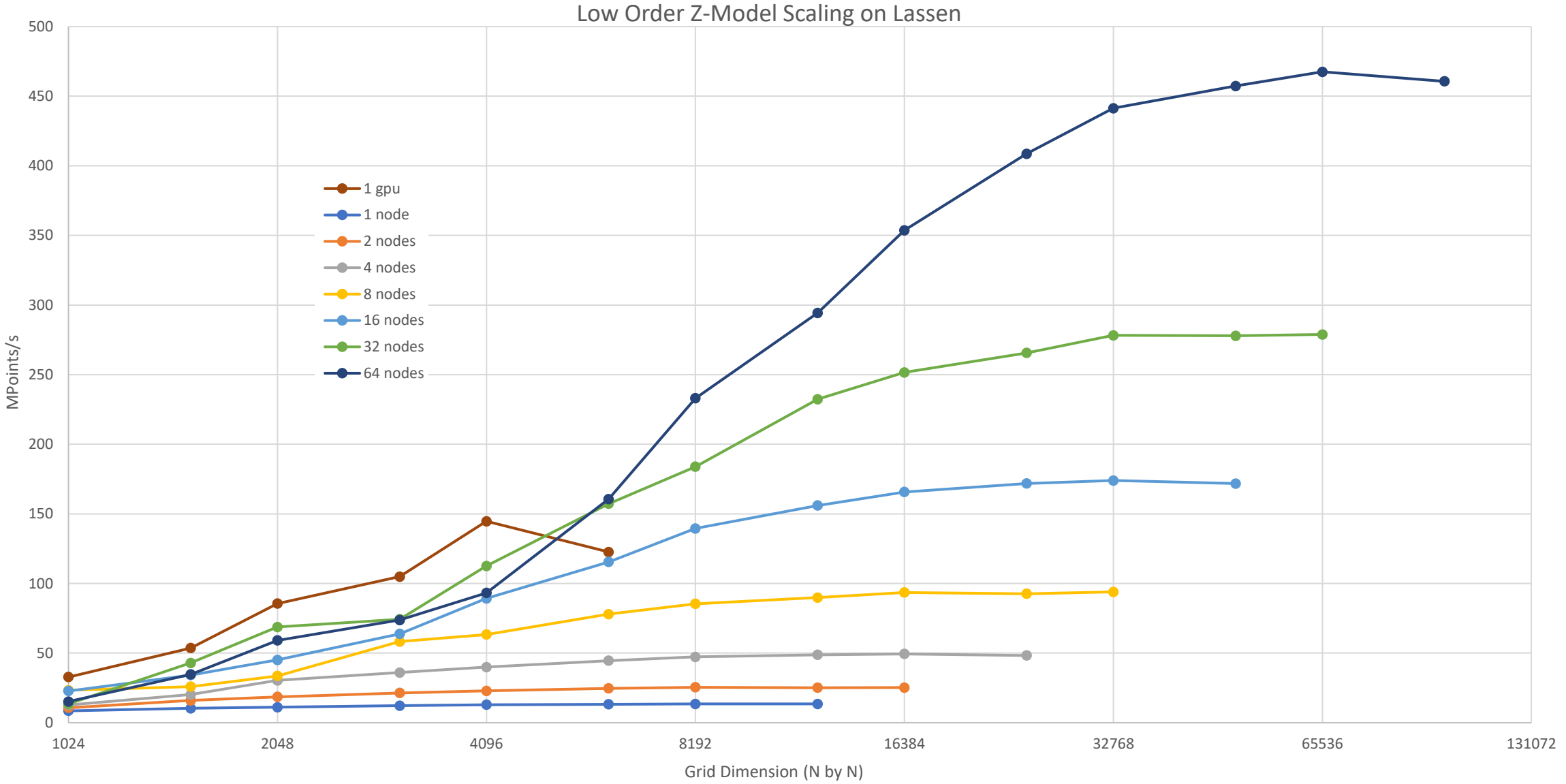
Why Z-Model?

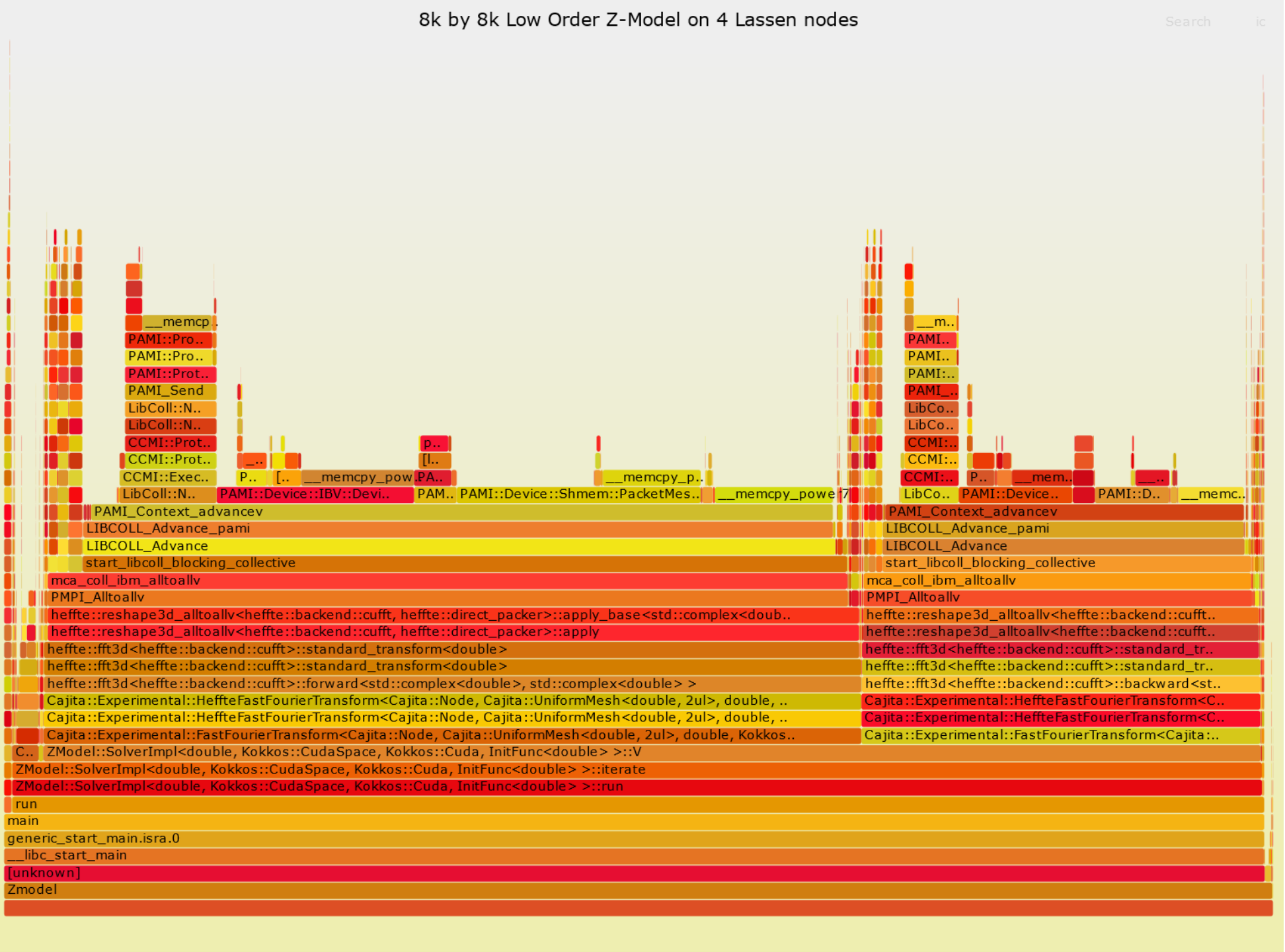
- We can look at global communication
- We can look at coupled codes
  - Combine Z-Model with a CFD solver

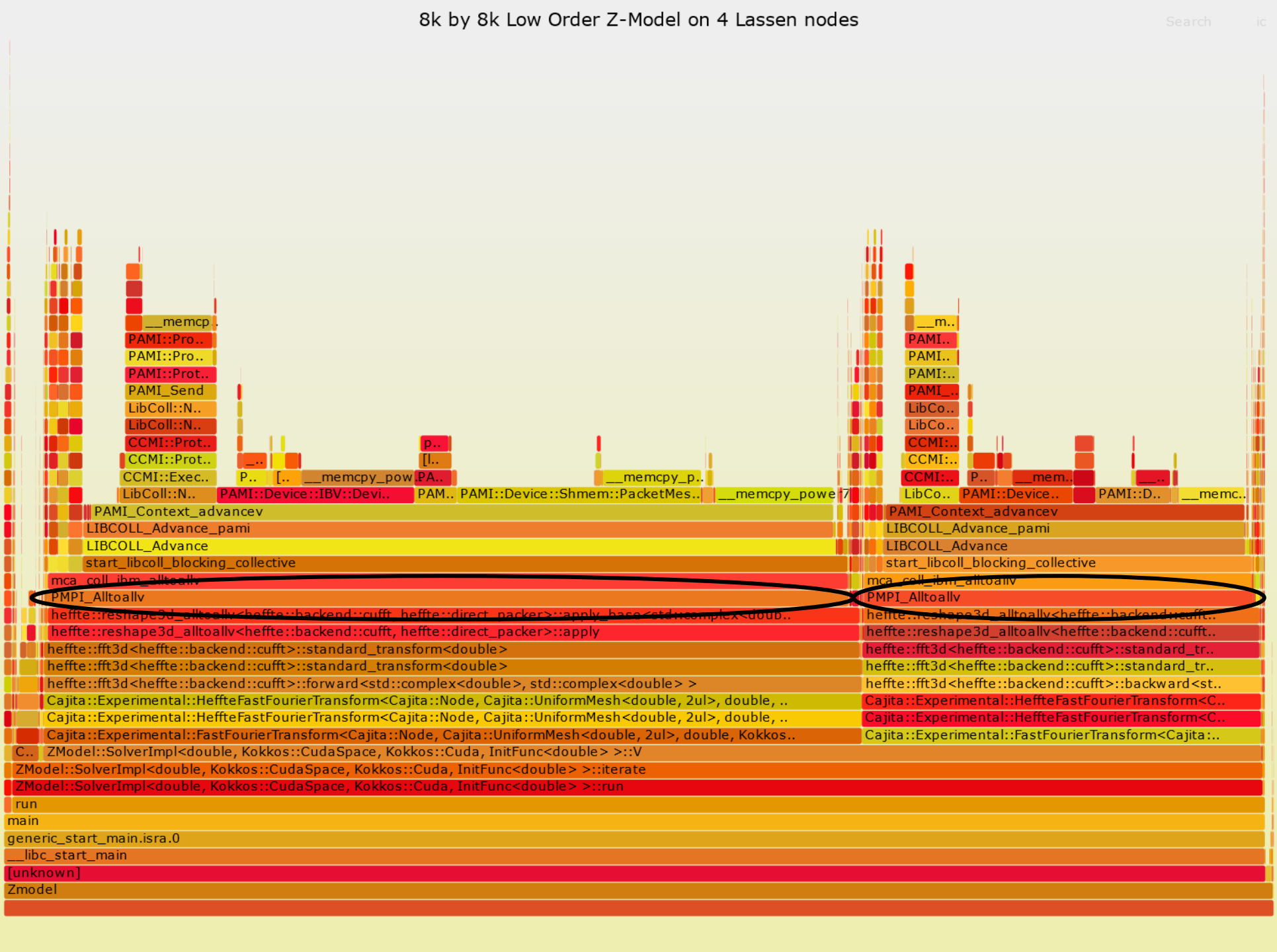
Low Order Z-Model Implementation

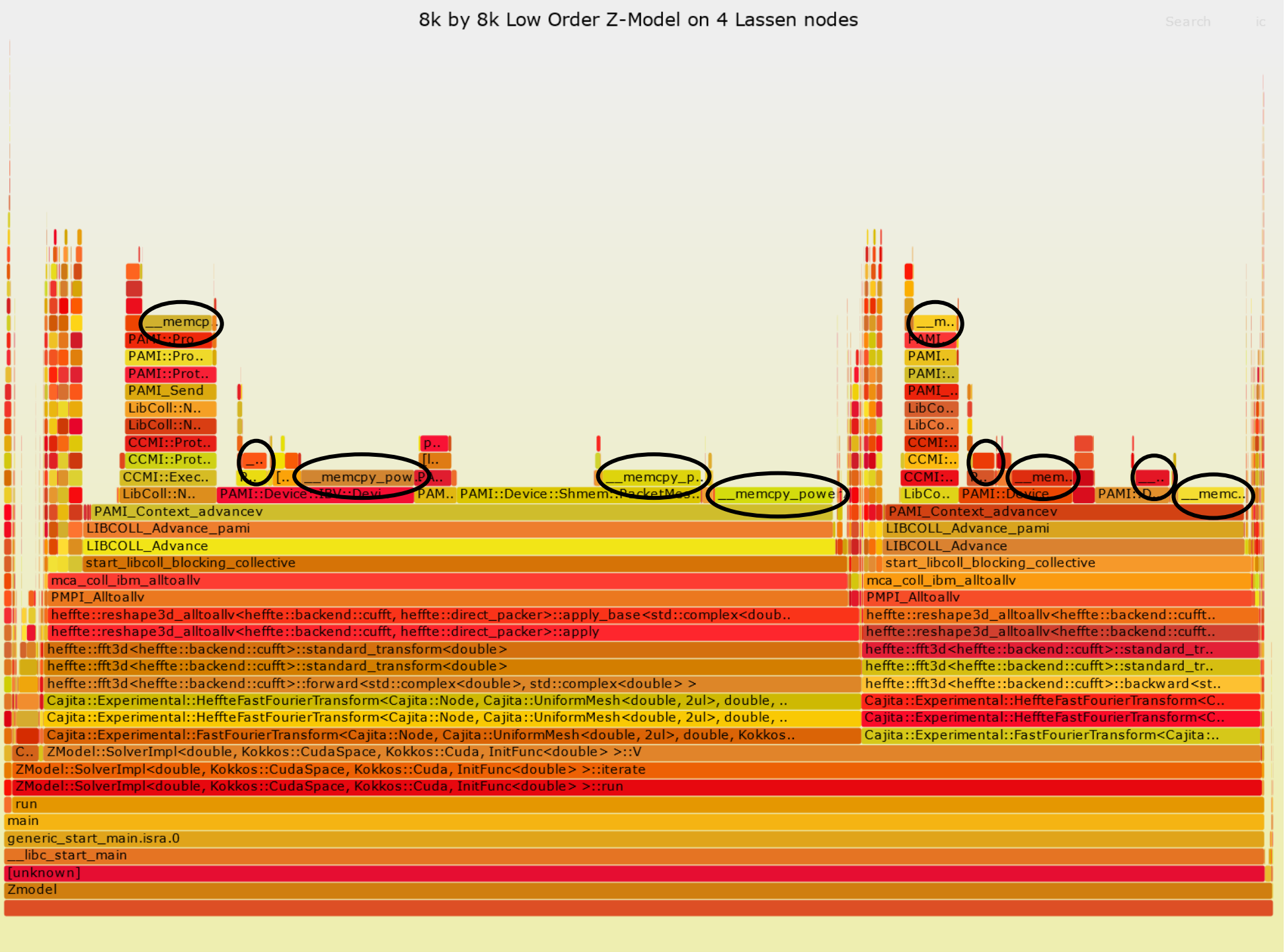
- Written in Kokkos/Cabana/Cajita
- Scales up to 98k x 98k mesh on 64 Lassen nodes (256 v100s)
- Runs at 450 Mpoints/s
- 170 iterations per hour with a 98k x 98k mesh


Low Order Z-Model Scaling on Lassen
MPoints/s
500
450
400
350
300
250
200
150
100
50
0
1 gpu
64 nodes
1024
2048
4096
8192
16384
32768
65536
131072
Grid Dimension (N by N)


Low Order Z-Model Scaling on Lassen
MPoints/s
Grid Dimension (N by N)
1 gpu
1 node
2 nodes
4 nodes
8 nodes
16 nodes
32 nodes
64 nodes
0
50
100
150
200
250
300
350
400
450
500
1024
2048
4096
8192
16384
32768
65536
131072


8k by 8k Low Order Z-Model on 4 Lassen nodes
Search
ic
__memcp..
PAMI::Pro..
PAMI::Pro..
PAMI::Prot..
PAMI_Send
LibColl::N..
LibColl::N..
CCMI::Prot..
CCMI::Prot..
CCMI::Exec..
LibColl::N..
P..
[..
__memcpy_pow..
PA..
p..
[l..
__memcpy_p..
PAM..
PAMI::Device::IBV::Devi..
PAMI::Device::Shmem::PacketMes..
__memcpy_powe..
__m..
PAMI..
PAMI..
PAMI:..
PAMI_..
LibCo..
LibCo..
CCMI:..
CCMI:..
CCMI:..
LibCo..
P..
__mem..
__..
PAMI::Device..
PAMI::D..
__memc..
PAMI_Context_advancev
LIBCOLL_Advance_pami
LIBCOLL_Advance
start_libcoll_blocking_collective
mca_coll_ibm_alltoallv
PMPI_Alltoallv
heffte::reshape3d_alltoallv<heffte::backend::cufft, heffte::direct_packer>::apply_base<std::complex<doub..
heffte::reshape3d_alltoallv<heffte::backend::cufft, heffte::direct_packer>::apply
heffte::fft3d<heffte::backend::cufft>::standard_transform<double>
heffte::fft3d<heffte::backend::cufft>::standard_transform<double>
heffte::fft3d<heffte::backend::cufft>::forward<std::complex<double>, std::complex<double> >
Cajita::Experimental::HeffteFastFourierTransform<Cajita::Node, Cajita::UniformMesh<double, 2ul>, double, ..
Cajita::Experimental::HeffteFastFourierTransform<Cajita::Node, Cajita::UniformMesh<double, 2ul>, double, ..
Cajita::Experimental::FastFourierTransform<Cajita::Node, Cajita::UniformMesh<double, 2ul>, double, Kokkos..
C..
ZModel::SolverImpl<double, Kokkos::CudaSpace, Kokkos::Cuda, InitFunc<double> >::V
ZModel::SolverImpl<double, Kokkos::CudaSpace, Kokkos::Cuda, InitFunc<double> >::iterate
ZModel::SolverImpl<double, Kokkos::CudaSpace, Kokkos::Cuda, InitFunc<double> >::run
run
main
generic_start_main.isra.0
__libc_start_main
[unknown]
Zmodel
heffte::reshape3d_alltoallv<heffte::backend::cufft..
heffte::reshape3d_alltoallv<heffte::backend::cufft..
heffte::fft3d<heffte::backend::cufft>::standard_tr..
heffte::fft3d<heffte::backend::cufft>::standard_tr..
heffte::fft3d<heffte::backend::cufft>::backward<st..
Cajita::Experimental::HeffteFastFourierTransform<C..
Cajita::Experimental::HeffteFastFourierTransform<C..
Cajita::Experimental::FastFourierTransform<Cajita:..


8k by 8k Low Order Z-Model on 4 Lassen nodes
Search
__memcp..
PAMI::Pro..
PAMI::Pro..
PAMI::Prot..
PAMI_Send
LibColl::N..
LibColl::N..
CCMI::Prot..
CCMI::Prot..
CCMI::Exec..
LibColl::N..
PAMI::Device::IBV::Devi..
PAM..
PAMI::Device::Shmem::PacketMes..
__memcpy_powe..
PAMI_Context_advancev
LIBCOLL_Advance_pami
LIBCOLL_Advance
start_libcoll_blocking_collective
mca_coll_ibm_alltoallv
PMPI_Alltoallv
heffte::reshape3d_alltoallv<heffte::backend::cufft, heffte::direct_packer>::apply
heffte::fft3d<heffte::backend::cufft>::standard_transform<double>
heffte::fft3d<heffte::backend::cufft>::standard_transform<double>
heffte::fft3d<heffte::backend::cufft>::forward<std::complex<double>, std::complex<double> >
Cajita::Experimental::HeffteFastFourierTransform<Cajita::Node, Cajita::UniformMesh<double, 2ul>, double, ..
Cajita::Experimental::HeffteFastFourierTransform<Cajita::Node, Cajita::UniformMesh<double, 2ul>, double, ..
Cajita::Experimental::FastFourierTransform<Cajita::Node, Cajita::UniformMesh<double, 2ul>, double, Kokkos..
ZModel::SolverImpl<double, Kokkos::CudaSpace, Kokkos::Cuda, InitFunc<double> >::V
ZModel::SolverImpl<double, Kokkos::CudaSpace, Kokkos::Cuda, InitFunc<double> >::iterate
ZModel::SolverImpl<double, Kokkos::CudaSpace, Kokkos::Cuda, InitFunc<double> >::run
run
main
generic_start_main.isra.0
__libc_start_main
[unknown]
Zmodel


8k by 8k Low Order Z-Model on 4 Lassen nodes
__memcp..
PAMI::Pro..
PAMI::Pro..
PAMI::Prot..
PAMI_Send
LibColl::N..
LibColl::N..
CCMI::Prot..
CCMI::Prot..
CCMI::Exec..
LibColl::N..
__memcpy_pow..
__memcpy_p..
__memcpy_powe..
PAMI::Device::Shmem::PacketMes..
PAMI_Context_advancev
LIBCOLL_Advance_pami
LIBCOLL_Advance
start_libcoll_blocking_collective
mca_coll_ibm_alltoallv
PMPI_Alltoallv
heffte::reshape3d_alltoallv<heffte::backend::cufft, heffte::direct_packer>::apply_base<std::complex<doub..
heffte::reshape3d_alltoallv<heffte::backend::cufft, heffte::direct_packer>::apply
heffte::fft3d<heffte::backend::cufft>::standard_transform<double>
heffte::fft3d<heffte::backend::cufft>::standard_transform<double>
heffte::fft3d<heffte::backend::cufft>::forward<std::complex<double>, std::complex<double> >
Cajita::Experimental::HeffteFastFourierTransform<Cajita::Node, Cajita::UniformMesh<double, 2ul>, double, ..
Cajita::Experimental::HeffteFastFourierTransform<Cajita::Node, Cajita::UniformMesh<double, 2ul>, double, ..
Cajita::Experimental::FastFourierTransform<Cajita::Node, Cajita::UniformMesh<double, 2ul>, double, Kokkos..
ZModel::SolverImpl<double, Kokkos::CudaSpace, Kokkos::Cuda, InitFunc<double> >::V
ZModel::SolverImpl<double, Kokkos::CudaSpace, Kokkos::Cuda, InitFunc<double> >::iterate
ZModel::SolverImpl<double, Kokkos::CudaSpace, Kokkos::Cuda, InitFunc<double> >::run
run
main
generic_start_main.isra.0
__libc_start_main
[unknown]
Zmodel

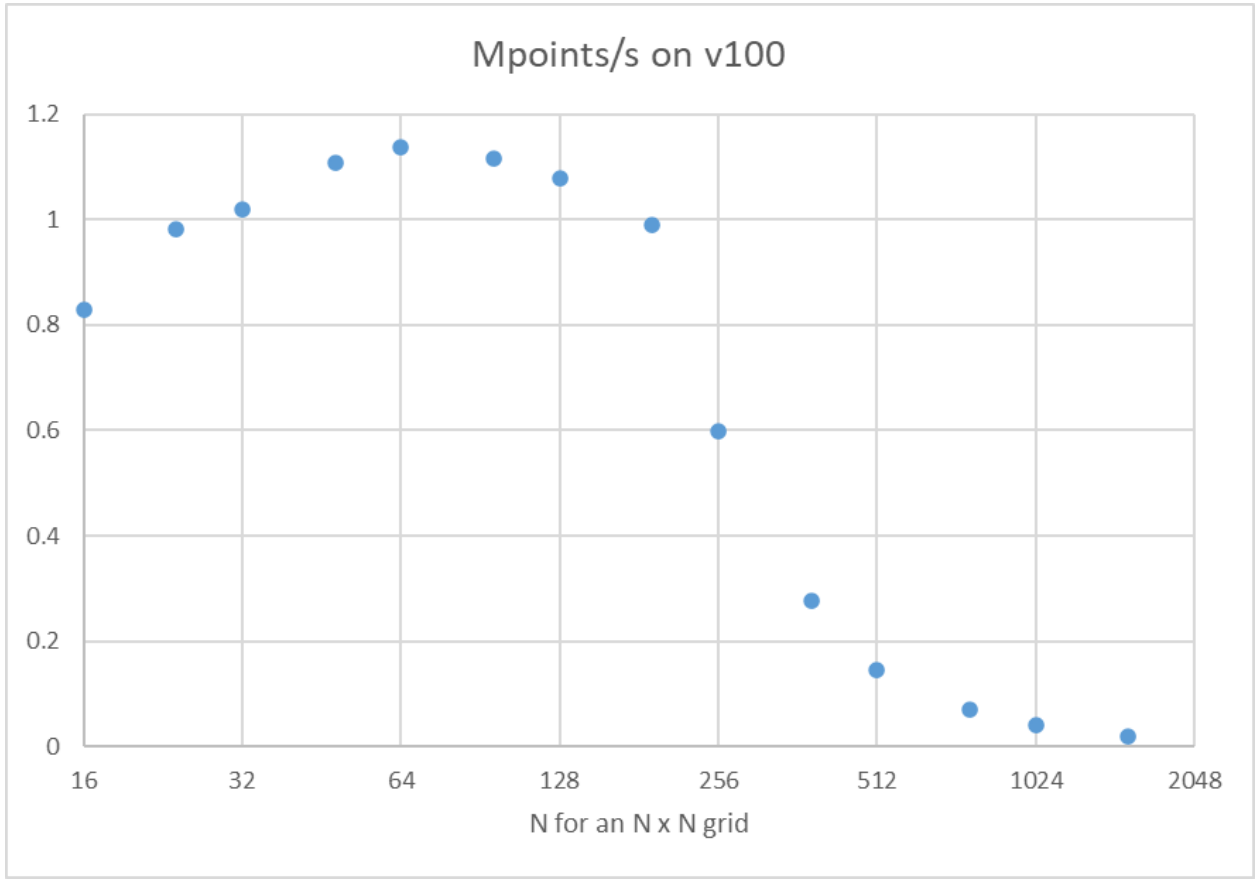
Low Order Z-Model Implementation

- Next Steps
  - Find source of mid-scale inefficiency
  - Working UTK and CMU about FFTs
  - Working with ECP-COPA (Stuart Slattery at ORNL) on FFTs in Cajita

High Order Z-Model Implementation

- Shared memory application written in Cuda
- Test program to figure out how to implement in distributed memory

High Order Z-Model Implementation



High Order Z-Model Implementation

- Bound by direct global forces calculation (O(n^4))
- At n=1024, 99.99% of the time is spent in global forces calculation
- Direct calculation does not scale
- Need approximate algorithms
  - Barnes-Hut
  - Fast Multipole Method (FMM)

High Order Z-Model Implementation

- Barnes-Hut or FMM lets us look at:
  - Hierarchical communication
  - Data migration
  - Global redistribution

Conclusion

- Parallelized the low order Z-Model
  - Looking into FFTs and global communication
- Parallelizing the high order Z-Model
  - Looking into Barnes-Hut/FMM and hierarchical communication