KAERI 한국원자력연구원 Korea Atomic Energy Research Institute

Massive Parallel Computation for an Efficient Whole Core Transport Calculation

2018.05.18
KAERI

Jin Young Cho and Seungsu Yuk

CONTENTS

Introduction

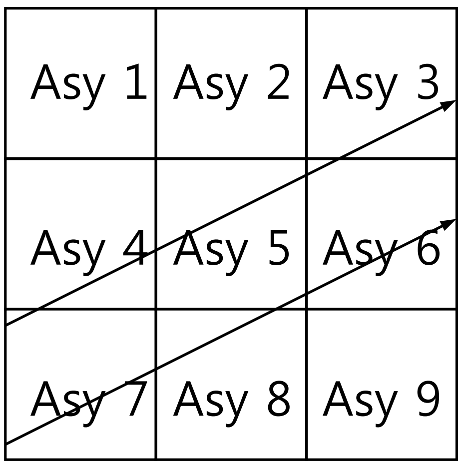
- DeCART Code
  - Sub-Pin Level Planar MOC Transport Calculation
    - Resonance Treatment based on Subgroup Method for Whole Core
    - Depletion Calculation based on Krylov Subspace Method
    - Double Heterogeneity Treatment by Sanchez Method
  - Pin Based Axial Solution for SP3 Transport Equation
  - CMFD Acceleration for Irregular Mesh
  - MPI based Parallel Computation
    - Apply to Axial Direction by Domain Decomposition
    - Plane-wise Distributed Memory
    - OpenMP to Radial Domain to Increase Computational Speed
    - Speedup Limit by the Number of Cores in a Node

Introduction

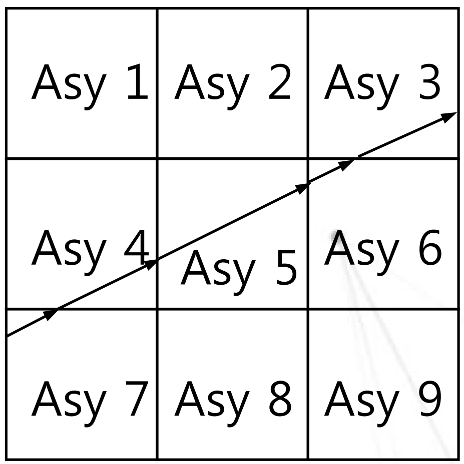
- MPI based Massive Parallel Computation
  - Applied to Axial and Radial Domain Decomposition
  - MPACT obtains a Good Computational Speed
  - The Least Domain Assigned to 1 Core can be 1 Planar Node
  - Applied to DeCART and Examine the Efficiency

Massive Parallelization

- **Modification in DeCART**
  - **Ray Tracing Scheme: Core Ray basis -> Assembly Ray basis**



Core Ray Based Ray Tracing

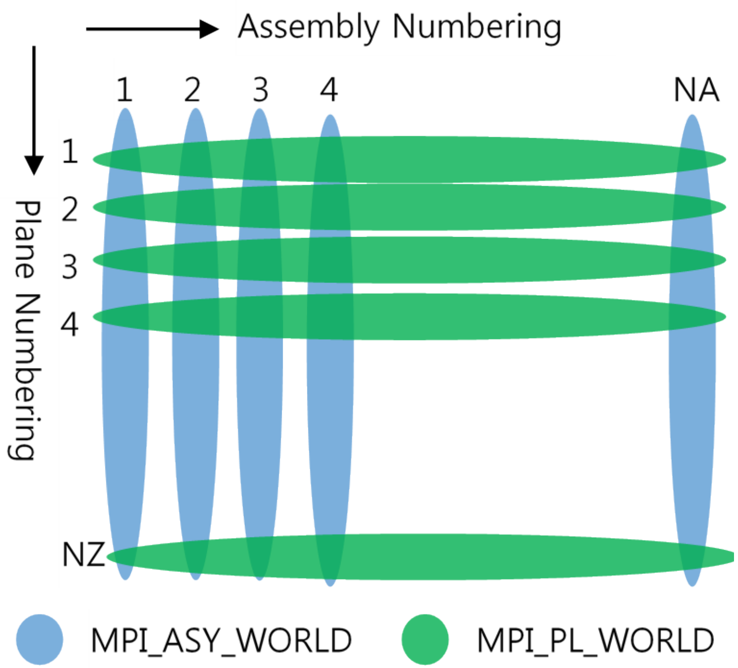


Assembly Ray Based Ray Tracing

- **Defect of Assembly Ray based Ray Tracing:**
  - Requires Assembly-wise Storage for Outgoing Angular Flux
  - Requires Communication with the Neighboring Assemblies to obtain the Incoming Angular Flux at the Assembly Boundary
  - Deteriorates Convergence Speed

Massive Parallelization

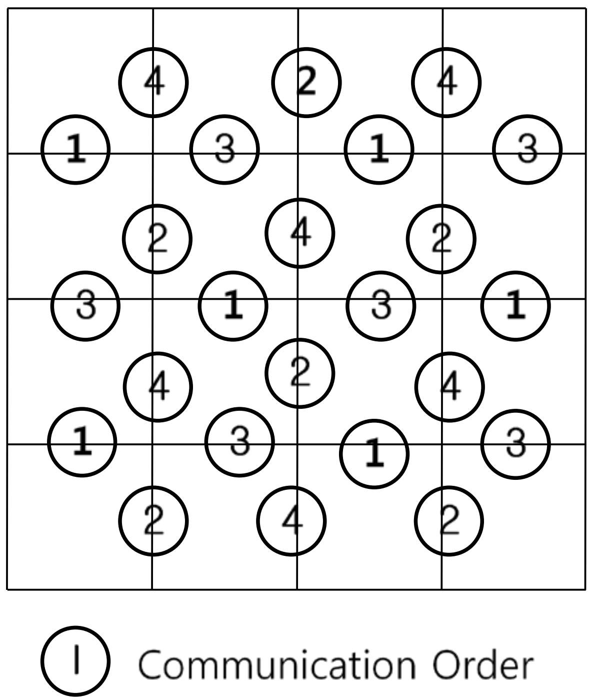
Communication Domain Splitting



- **MPI_COMM_WORLD Splits into Multiple MPI_ASY_WORLDs and MPI_PL_WORLDs for an Efficient Processor Control**
- **MPI_PL_WORLD**
  - Consists of Processors solving the Same Plane
  - Communication: When Solving Radial MOC and the Whole CMFD Calculations
- **MPI_ASY_WORLD**
  - Consists of Processors solving the Same Assembly
  - Communication: When Solving Axial Pn and the Whole CMFD Calculations

Massive Parallelization

Communication in MPI_PL_WORLD

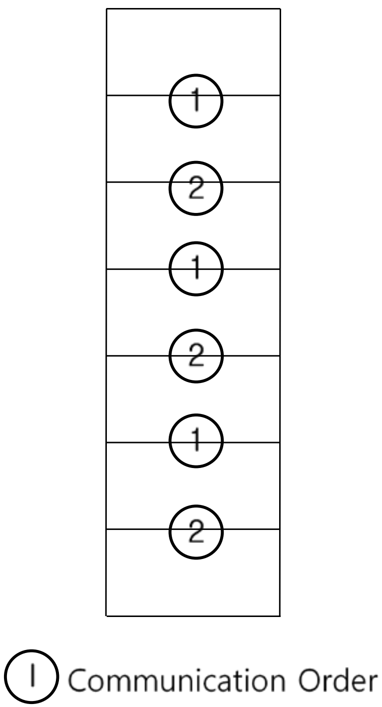


- Maximum 4 Communications with Neighboring Processors are Required for Incoming Angular Flux
- Communication Order
  - Index = I+J

    I,J: Assembly Coordinates (I,J)
  - Even Index: Clockwise from Bottom Surface, and Receives Data First and Sends Next
  - Odd Index: Clockwise from Top Surface, and Sends Data First and Receives Next
- Data Communication Occurs Simultaneously According to the Communication Order

Massive Parallelization

Communication in MPI_ASY_WORLD



- **Maximum 2 Communications with Neighboring Processors are Required for Flux Moments in Pn Solver or Scalar Flux in CMFD Solver**
- **Communication Order**
  - Index = Plane Number
  - Odd Index: Communicates with Lower Plane first and next with Upper Plane, and Receives Data First and Sends Next
  - Even Index: Communicates with Upper Plane first and next with Lower Plane, and Sends Data First and Receives Next
- **Data Communication Occurs Simultaneously According to the Communication Order**

Performance Examination

- Test Problems
  - C5G7 2D Problem
  - C5G7 3D Problem
  - SMART 3D Problem
- Calculation Model
  - 0.02 cm Ray Spacing, 32 Gaussian Quadrature Azimuthal Angles and 2 Optimum Polar Angles
  - P2 Anisotropic Scattering Angle Treatment
  - 10 Sub-planes per MOC Plane for Axial P3 Transport Calculation using LPEN Method

Performance Examination – C5G7 2D Problem

Computational Efficiency

	Size, Np	Total	CMFD	MOC
Time, sec	Fuel Assembly (FA), 1	16.53	0.11	16.16
	Octant Core (OC), 6	17.97	0.21	17.37
	Quarter Core (QC), 9	17.91	0.23	17.35
	Full Core (FC), 36	18.14	0.37	17.25
$\eta = T_{FA}/T$	Octant Core (OC), 6	0.92	0.52	0.93
	Quarter Core (QC), 9	0.92	0.48	0.93
	Full Core (FC), 36	0.91	0.30	0.94

- **Reference Computing Time : Computing Time for Single Fuel Assembly**
- **Full Core Problem: 36 Processors are Attended**
- **Poor Efficiency in CMFD: Due to the Frequent Communication**

Performance Examination – C5G7 3D Problem

Computational Efficiency

	Size, Np	Total	CMFD	Pn	MOC
Time, sec	Fuel Assembly (FA), 4	19.37	1.63	1.37	17.46
	Octant Core (OC), 24	19.72	2.61	2.12	16.70
	Quarter Core (QC), 36	20.50	2.63	2.09	17.50
	Full Core (FC), 144	22.42	3.61	2.72	17.96
$\eta = T_{FA}/T$	Octant Core (OC), 24	0.98	0.62	0.65	1.05
	Quarter Core (QC), 36	0.94	0.62	0.66	1.00
	Full Core (FC), 144	0.86	0.45	0.50	0.97

- **Reference Computing Time : Computing Time for Single Fuel Assembly with 4 Processors for Axial Pn Calculation**
- **Full Core Problem: 144 Processors are Attended**
- **Over Efficiency in MOC for QC Problem: Due to the Lower Burden in MASTER Processor**

Performance Examination – C5G7 3D Problem

KAERI 한국원자력연구원 Korea Atomic Energy Research Institute

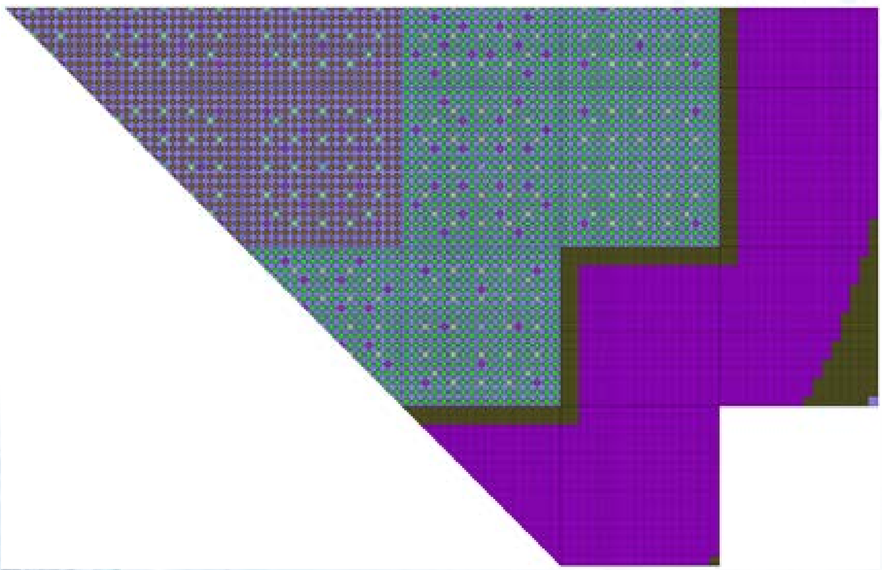
Computing Time using MPI + OpenMP Scheme

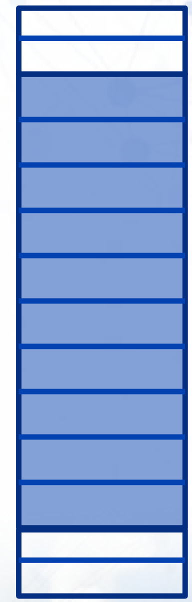
Size	Threads	Total	CMFD	Pn	MOC
Octant Core (OC)	12	13.94	2.62	1.67	9.51
	6	21.18	3.75	2.67	15.58
Quarter Core (QC)	12	30.23	6.08	3.90	18.74
	9	35.60	6.49	4.30	23.79
Full Core (FC)	12	133.29	26.08	16.62	61.08

- The Number of Threads 12 is the Maximum Number available in Current Cluster Configuration
- Use of 6 Threads for OC and 9 for QC are for Efficiency Comparison with MPI Based Domain Decomposition Scheme
- OpenMP shows Worse Efficiency than MPI

Performance Examination – SMART 3D Core

Octant Core





- 17 Assemblies with 11 Fuel and 6 Reflector Assemblies
- 14 Planes with 10 Fuel and 4 Reflector Planes

Performance Examination – SMART 3D Core

KAERI 한국원자력연구원 Korea Atomic Energy Research Institute

Computational Efficiency

	Size	Total	CMFD	Pn	MOC
Time, sec	Fuel Assembly (FA)	337.8	12.1	8.8	321.0
	Octant Core (OC)	420.3	37.7	25.6	356.7
η	Octant Core (OC)	0.80	0.32	0.34	0.90

$$\eta = T_{FA}/T$$

- Reference Computing Time : Computing Time for Single Fuel Assembly with 14 Processors for Axial Pn Calculation
- Octant Core Problem: 238 Processors are Attended
- MOC Calculation Requires about 85 % of Total Computing Time

Conclusion

- In this study, Massive Parallel Computation is introduced to Whole Core Transport Calculation, and the Performance is Examined for C5G7 2-D/3-D Problems and for the Realistic 3-D Core Problem
- Parallel Efficiency for Total Computing Time

Problems	C5G7 2-D			C5G7 3-D			SMART
	OC	QC	FC	OC	QC	FC	OC
η	0.92	0.92	0.91	0.98	0.94	0.86	0.80

➡ Good Efficiency Considering the Number of Cores

- Therefore, it can be concluded that the massive parallel computation works good for the whole core transport calculation
- In the Parallel Efficiency, the CMFD and Axial Pn Calculation showed Poor Efficiency, and deteriorates the Total Efficiency. Therefore, In the Future, Improvement in Calculation Algorithm is needed to be Developed for Parallel Efficiency