Lecture 10: HPC for Finite Arrays

ANSYS HFSS for Antenna Design
Finite Array Example

• Model
  • This example is intended to demonstrate how to use HFSS to perform finite array analysis using HFSS’s finite array domain decomposition method (DDM)

• Before you start:
  • You will need to complete the previous workshop: Unit Cell
Antenna Arrays

- **Phased Array**
  - A group of antenna elements in which the relative amplitudes and phases are varied to construct an effective radiation pattern by constructive and destructive interference

\[
E_{array}(\theta_o, \phi_o, \theta, \phi) = \sum_n A_n(\theta_o, \phi_o)e^{j\psi_n(\theta_o,\phi_o)} \frac{e^{-jk_or_n}}{r_n} E_n(\theta, \phi)
\]

\[
S_m(\theta_o, \phi_o) = \sum_n \frac{A_n(\theta_o, \phi_o)e^{j\psi_n(\theta_o,\phi_o)}}{A_m(\theta_o, \phi_o)e^{j\psi_m(\theta_o,\phi_o)}} S_{m,n}
\]

- Beam shape can be controlled by adjusting the amplitude of each element.
- Beam can be steered by applying a progressing phase shift across the array.
- Mutual coupling plays a key role in an element’s pattern and input impedance.
- It is necessary to analyze the arrays performance over frequency and scan volume.
Antenna Arrays and the Unit Cell

**Unit Cell**

- Uses Master/Slave boundaries
  - Models a single element as if it were in an infinite array environment
  - Infinite array environment accounted for by enforcing field periodicity through master/slave boundary pairs
- Reduces RAM
- Reduces solve time

**Infinite Array Approx.**
- Edge affects ignored
- Uniform magnitude excitation
- Single scan angle solved at a time (Distributed Solve Option Parallelizes)

**Finite Array**

**Explicit**

- Entire array analyzed
  - Accounts for edge affects and edge treatments
  - Provides mutual coupling terms
  - Allows magnitude taper

- Most flexible
  - Fewest assumptions
  - Adaptive meshing performed on entire model

- Complex Geometry
  - Every element needs to be drawn
  - Large number of excitations
  - Complicated meshing process

**Finite Array DDM**

- Uses Master/Slave boundaries
- Entire array analyzed
  - Accounts for edge affects
  - Provides mutual coupling terms
  - Allows magnitude taper
  - Adaptive meshing performed on single unit cell
- Distributes RAM
- Reduces solve time
- Periodic assumption
  - Geometry must be purely periodic in the XY plane
Domain Decomposition Overview

- DDM distributes the mesh and accesses distributed RAM throughout a network
  - DDM was first released in HFSSv12
  - Distributes a model’s mesh/solution across several computers distributing the RAM
  - Solves a model’s full behavior as if run on a single computer

Generalized DDM

- Addresses:
  - RAM limitations
- Does not address:
  - Meshing time
  - Complex geometry issues
Solution: Finite Array Domain Decomposition

• Utilizes Replicated DDM Unit Cell to Address Array Concerns

• Geometry and Mesh copied directly from Unit Cell Model
  • Unit Cell geometry expanded to finite array through a simple GUI
  • Adaptive Meshing Process imported from Unit Cell Simulation
    – Dramatically reduces the meshing time associated with finite array analyses.
    – Mesh periodicity reinforces array’s periodicity.
  • Full Array Mesh Solved Using DDM

Addresses:
• RAM limitations
• Meshing time
• Complex geometry issues
Finite Array DDM Tool Advantages

- Advantages of the finite array DDM:
  - Solves much **BIGGER** arrays on the same hardware
  - Obtains **ACCURATE** results that match HFSS explicit simulations
  - Enables **EFFICIENT** simulation of large finite arrays utilizing domain decomposition (DDM)
  - Makes it **EASY** to transform a master/slave unit cell into a finite array

Embedded element pattern:
Finite Array DDM vs. Explicit

256 element Vivaldi with metal thickness, under 32GB of RAM!
How It Works!!!

- Start off with a Unit Cell Model to create the Unit Cell Mesh
  - Accounts for infinite array behavior
  - Unit Cell Simulation is Fast
  - Unit Cell Simulation is Memory Efficient
  - May use radiation boundary, PML or FE-BI on top surface of unit cell

Radiation Boundary or PML Absorbs Radiated Fields

Master / Slave Boundary Pairs Mimic Array’s Periodicity
How It Works!!!

- **Construct the Finite Array**
  - Simple GUI based creation
  - Reduces model complexity
  - Reduces display issues
  - Creates 1 unit cell of additional space around the edge elements to terminate the array fields in vacuum or infinite ground plane.

Defines directions of periodicity based on Master / Slave Boundaries

Defines number of elements in each direction
Finite Array DDM vs Explicit: Simulation Time/RAM

<table>
<thead>
<tr>
<th></th>
<th>Simulation Time</th>
<th>RAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finite Array DDM</td>
<td>2.75hrs</td>
<td>10.5GB</td>
</tr>
<tr>
<td>Explicit</td>
<td>6hrs</td>
<td>28GB</td>
</tr>
</tbody>
</table>

Note: May observe greater RAM and time savings with complex models!

**2.2X SPEEDUP**

**REDUCED RAM REQUIREMENT**
Finite Array DDM vs Explicit: Results Comparison

- Finite Array DDM provides the same results more efficiently
This page intentionally left blank