Altair Feko 2022.1.1

Example Guide

Updated: 08/12/2022

Intellectual Property Rights Notice

Altair® ultraFluidX® ©2010-2022

Altair® Virtual Wind Tunnel™ ©2012-2022

Altair® WinProp™ ©2000-2022

Altair® WRAP™ ©1998-2022

Altair® S-FRAME® ©1995-2022

Altair® S-STEEL™ ©1995-2022

Altair® S-PAD™ ©1995-2022

Altair® S-CONCRETE™ ©1995-2022

Altair® S-LINE™ ©1995-2022

Altair® S-TIMBER™ ©1995-2022

Altair® S-FOUNDATION™ ©1995-2022

Altair® S-CALC™ ©1995-2022

Altair® Structural Office™ ©2022

Altair Packaged Solution Offerings (PSOs)

Altair® Automated Reporting Director™ ©2008-2022

Altair® e-Motor Director™ ©2019-2022

Altair® Geomechanics Director™ ©2011-2022

Altair® Impact Simulation Director™ ©2010-2022

Altair® Model Mesher Director™ ©2010-2022

Altair® NVH Director™ ©2010-2022

Altair® NVH Full Vehicle™ ©2022

Altair® NVH Standard™ ©2022

Altair® Squeak and Rattle Director™ ©2012-2022

Altair® Virtual Gauge Director™ ©2012-2022

Altair® Weld Certification Director™ ©2014-2022

Altair® Multi-Disciplinary Optimization Director™ ©2012-2022

Altair HPC & Cloud Products

Altair® PBS Professional® ©1994-2022

Altair® PBS Works™ ©2022

Altair® Control™ ©2008-2022

Altair ®Access™ ©2008-2022

Altair® Accelerator™ ©1995-2022

Altair® Accelerator™ Plus©1995-2022

Altair® FlowTracer™ ©1995-2022

Altair® Allocator™ ©1995-2022

Altair® Monitor™ ©1995-2022

Altair® Hero™ ©1995-2022

Altair® Software Asset Optimization (SAO) ©2007-2022

Altair Mistral™ ©2022

Altair Drive ©2021-2022

Altair® Grid Engine® ©2001, 2011-2022

Altair® DesignAI™ ©2022

Altair Breeze™ ©2022

Altair® NavOps® ©2022

Altair® Unlimited™ ©2022

Altair Data Analytics Products

Altair Analytics Engine™ ©2002-2022

Altair Analytics Hub ©2002-2022

Altair Analytics Workbench™ ©2002-2022

Altair® Knowledge Studio® ©1994-2022

Altair® Knowledge Studio® for Apache Spark ©1994-2022

Altair® Knowledge Seeker™ ©1994-2022

Altair® Knowledge Hub™ ©2017-2022

Altair® Monarch® ©1996-2022

Altair® Panopticon™ ©2004-2022

Altair® SmartWorks™ ©2021-2022

Altair SmartCore™ ©2011-2022

Altair SmartEdge™ ©2011-2022

Altair SmartSight™ ©2011-2022

Altair® License Utility™ ©2010-2022

Altair One™ ©1994-2022

2022.1

May 12, 2022

Technical Support

Altair provides comprehensive software support via web FAQs, tutorials, training classes, telephone, and e-mail.

Altair One Customer Portal

Altair One (https://altairone.com/) is Altair's customer portal giving you access to product downloads, a Knowledge Base, and customer support. We recommend that all users create an Altair One account and use it as their primary portal for everything Altair.

When your Altair One account is set up, you can access the Altair support page via this link: www.altair.com/customer-support/

Altair Community

Participate in an online community where you can share insights, collaborate with colleagues and peers, and find more ways to take full advantage of Altair's products.

Visit the Altair Community (https://community.altair.com/community) where you can access online discussions, a knowledge base of product information, and an online form to contact Support. These valuable resources help you discover, learn and grow, all while having the opportunity to network with fellow explorers like yourself.

Altair Training Classes

Altair's in-person, online, and self-paced trainings provide hands-on introduction to our products, focusing on overall functionality. Trainings are conducted at our corporate and regional offices or at your facility.

For more information visit: https://learn.altair.com/

If you are interested in training at your facility, contact your account manager for more details. If you do not know who your account manager is, contact your local support office and they will connect you with your account manager.

Telephone and E-mail

If you are unable to contact Altair support via the customer portal, you may reach out to technical support via phone or e-mail. Use the following table as a reference to locate the support office for your region.

When contacting Altair support, specify the product and version number you are using along with a detailed description of the problem. It is beneficial for the support engineer to know what type of workstation, operating system, RAM, and graphics board you have, so include that in your communication.

Location	Telephone	E-mail
Australia	+61 3 9866 5557	anzsupport@altair.com
Brazil	+55 113 884 0414	br_support@altair.com

Location	Telephone	E-mail
Canada	+1 416 447 6463	support@altairengineering.ca
China	+86 400 619 6186	support@altair.com.cn
France	+33 141 33 0992	francesupport@altair.com
Germany	+49 703 162 0822	hwsupport@altair.de
Greece	+30 231 047 3311	eesupport@altair.com
India	+91 806 629 4500 +1 800 425 0234 (toll free)	support@india.altair.com
Israel		israelsupport@altair.com
Italy	+39 800 905 595	support@altairengineering.it
Japan	+81 3 6225 5830	support@altairjp.co.jp
Malaysia	+60 32 742 7890	aseansupport@altair.com
Mexico	+52 55 5658 6808	mx-support@altair.com
New Zealand	+64 9 413 7981	anzsupport@altair.com
South Africa	+27 21 831 1500	support@altair.co.za
South Korea	+82 704 050 9200	support@altair.co.kr
Spain	+34 910 810 080	support-spain@altair.com
Sweden	+46 46 460 2828	support@altair.se
United Kingdom	+44 192 646 8600	support@uk.altair.com
United States	+1 248 614 2425	hwsupport@altair.com

If your company is being serviced by an Altair partner, you can find that information on our web site at https://www.altair.com/PartnerSearch/.

See www.altair.com for complete information on Altair, our team, and our products.

Contents

Antenna Synthesis and Analysis

A

Simple examples demonstrating antenna synthesis and analysis.

This chapter covers the following:

A.1 Dipole

Calculate the radiation pattern and input impedance for a half-wavelength dipole at 74.9 MHz. The dipole length is 2 m with a wire radius of 2 mm.

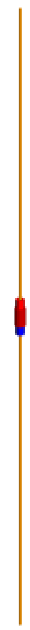


Figure 1: A 3D view of the dipole with a voltage source in CADFEKO.

A.1.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
   - *lambda* = 4 (The wavelength in free space.)
   - *freq* = *c0*/*lambda* (The operating frequency.)
   - *h* = *lambda*/2 (Length of the dipole.)
   - *radius* = 2E-3 (Radius of the wire.)
2. Create the dipole.
   a) Create a line.
      - **Start point**: (0, 0, -*h*/2)
      - **End point**: (0, 0, *h*/2)

   b) Add a wire port to the middle of the line.

   c) Add a voltage source to the port. (1 V, 0°, 50 Ω).
3. Set the frequency to *freq*.

A.1.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a vertical far field request (-180°≤θ≤180°, with ϕ=0°). Sample the far field at θ=2° steps.

A.1.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to *radius*.

A.1.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.1.5 Viewing the Results

View and post-process the results in POSTFEKO.

1. View the gain (in dB) of the requested far field pattern using a polar plot.
   a) On the **Display** tab, in the **Axes** group, click **Axis settings**, and then click the **Radial** tab. Set **Maximum dynamic range in dB** to `10` dB.

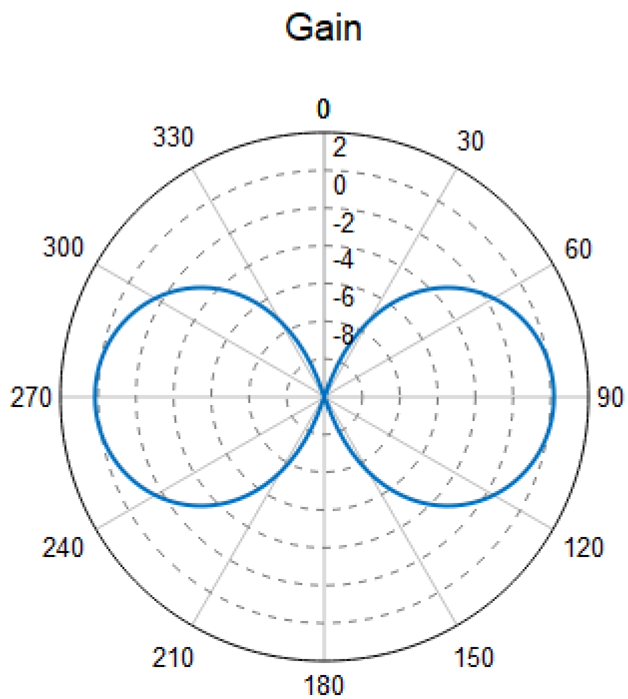


Figure 2: A polar plot of the requested far field gain (dB) viewed in POSTFEKO.

2. Review the impedance at a single frequency using one of the following methods:
- Plot the impedance as a function of frequency on a Cartesian graph or Smith chart.
- View the impedance value in the `*.out` file. Open the `.out` file in the output file viewer (in POSTFEKO), or in any other text file viewer.

```
                         DATA OF THE VOLTAGE SOURCE NO. 1

                   real part  imag. part   magnitude       phase
Current    in A    1.0144E-02 -4.9861E-03  1.1303E-02      -26.18
Admitt.    in A/V  1.0144E-02 -4.9861E-03  1.1303E-02      -26.18
Impedance in Ohm   7.9398E+01  3.9027E+01  8.8471E+01       26.18
Inductance  in H   8.2875E-08
```

A.2 Dipole in Front of a Cube

Calculate the radiation pattern for a half-wavelength dipole in front of a cuboid. View the effect of the cuboid on the radiation pattern.

The far field results are compared for the following cuboid configurations:

1. Perfect electric conductor (PEC) cuboid
2. Lossy metallic cuboid
3. Dielectric cuboid

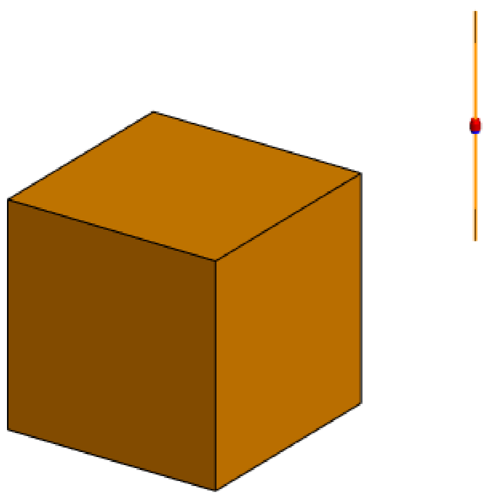


Figure 3: A 3D view of the dipole in front of a PEC cuboid in CADFEKO.

> **Tip:** Each model uses its predecessor as a starting point. Create the models in their presentation order. Save each model to a new location to keep them.

A.2.1 Dipole and PEC Cube

Create the dipole and cuboid. Model the cuboid using PEC.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
   - *lambda* = 4 (The wavelength in free space.)
   - *freq* = *c0*/*lambda* (The operating frequency.)
   - *h* = *lambda*/2 (Length of the dipole.)
   - *radius* = 4e-3 (Radius of the wire.)
2. Define the following named points:
   - *cuboidCorner*: (0, *-lambda/4*, *-lambda/4*)
   - *lineEnd*: (0, 0, *h*/2)
   - *lineStart*: (0, 0, -*h*/2)
   - *offset*: (-3**lambda*/4, 0, 0)
3. Create a cube.
   a) Create a cuboid. By default the cuboid is PEC.
      - **Definition method**: **Base corner, width, depth, height**
      - **Base corner**: (*cuboidCorner*, *cuboidCorner*, *cuboidCorner*)
      - **Width (W)**: *lambda*/2
      - **Depth (D)**: *lambda*/2
      - **Height (H)**: *lambda*/2
4. Create the dipole.
   a) Create a line.
      - **Start point**: (*lineStart*, *lineStart*, *lineStart*)
      - **End point**: (*lineEnd*, *lineEnd*, *lineEnd*)

   b) Translate the line in the negative X direction from (0, 0, 0) to the named point, *offset*.
   c) Add a wire port to the middle of the line.
   d) Add a voltage source to the port. (1 V, 0°, 50 Ω).
5. Set the frequency to *freq*.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

View the distortion in the radiation pattern of the dipole due to the proximity of the cuboid.
Create a horizontal far field request (0°≤ϕ≤360°, with θ=90°). Sample the far field at ϕ=2° steps.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to *radius*.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.2.2 Dipole and Lossy Metal Cube

Create the dipole and cuboid. Model the cuboid using lossy metallic surfaces and the region inside the cuboid as a vacuum.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Dipole and PEC Cube and rename the file.
2. Create a metallic medium.
   a) **Label**: `lossy_metal`
   b) **Conductivity**: `1e2`
3. Set the region inside the cuboid to **Free space**.

   > **Tip:** Open the **Modify Region** dialog and click the **Properties** tab. From the **Medium** list select **Free space**.

4. Change the cuboid faces to *lossy_metal* and set the thickness to `0.005`.

   > **Tip:** Open the **Modify Face** dialog and click the **Properties** tab. From the **Medium** list select **lossy_metal**.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Dipole and PEC Cube model.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for Dipole and PEC Cube model.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.2.3 Dipole and Dielectric Cube

Create the dipole and cuboid. Model the cuboid as a dielectric solid.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Dipole and Lossy Metal Cube and rename the file.
2. Create a dielectric medium.
   a) **Label**: `diel`
   b) **Relative permittivity**: `2`
3. Set the region inside the cuboid to *diel*.
4. Set the face media for all faces of the cuboid to **Default**.

   > **Note:** When the **Default** face medium is specified, CADFEKO selects the face medium based on the region setting.

5. Delete the `lossy_metal` medium.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests from the Dipole and PEC Cube model.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for Dipole and PEC Cube model.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.2.4 Viewing the Results

View and post-process the results in POSTFEKO.

Compare the gain (in dB) of the requested far field patterns on a polar plot.

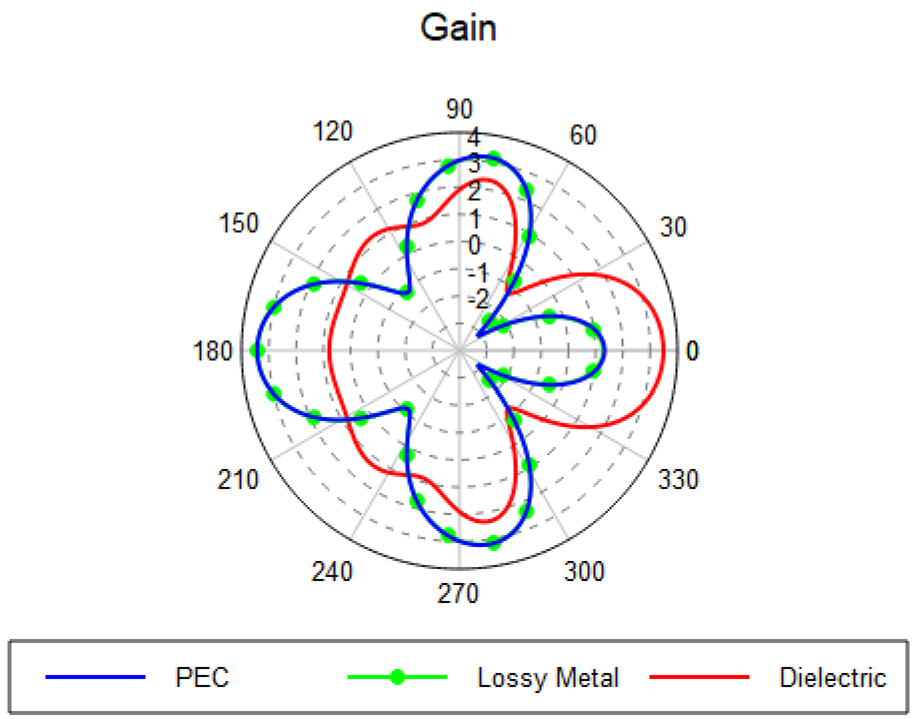


Figure 4: A polar plot of the requested radiation patterns for the dipole and cuboid configurations in POSTFEKO.

Note: View the pronounced scattering effect of the cuboid on the dipole radiation pattern. The dielectric cube results in a gain increase in the direction of the cube.

A.3 Dipole in Front of a Plate

Calculate the radiation pattern of a dipole in front of an electrically large plate. Several techniques available in Feko are considered and the results and resource requirements compared.

The radiation pattern is compared for the following techniques:

1. Method of moments (MoM)
2. Method of moments with higher order basis functions (HOBF)
3. Finite difference time domain (FDTD)
4. Uniform theory of diffraction (UTD)
5. Ray-launching geometrical optics (RL-GO)
6. Physical optics (PO)
7. Large element physical optics (LE-PO)

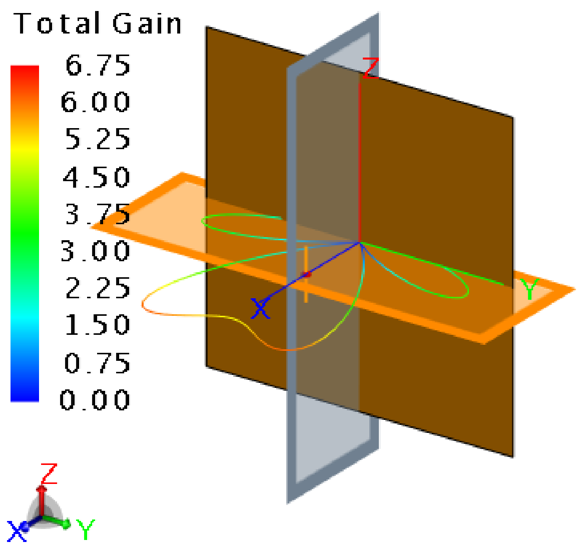


Figure 5: A 3D view of the dipole with a metallic plate.

Tip: Each model uses its predecessor as a starting point. Create the models in their presentation order. Save each model to a new location to keep them.

A.3.1 Dipole in Front of a Plate with MoM

Create the dipole and the rectangular plate. Solve the model using the method of moments (MoM).

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
    - *d* = 2.25 (Distance between dipole and plate. [3*lambda/4])
    - *h* = 1.5 (Length of the dipole. [lambda/2])
    - *a* = 4.5 (Half the plate length.)
    - *rho* = 0.03 (Radius of the wire.)
2. Create the dipole.
    a) Create a line.
        - **Start point**: (*d*, 0, -*h*/2)
        - **End point**: (*d*, 0, *h*/2)
3. Create the plate.
    a) Create a rectangle.
        - **Definition method**: **Base centre, width, depth**
        - **Base centre (C)**: (0, 0, 0)
        - **Width (W)**: 2**a*
        - **Depth (D)**: 2**a*
        - **Custom workplane**:
            - **Origin**: (0, 0, 0)
            - **U vector**: (0, 0, -1)
            - **V vector**: (0, 1, 0)
4. Add a wire port to the middle of the line.
5. Add a voltage source to the port. (1 V, 0°, 50 Ω).
6. Set the **Frequency** to *c0*/3.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

View the distortion in the dipole radiation pattern due to proximity of the plate.
Create a horizontal far field request (0°≤ϕ≤360°, with θ=90°). Sample the far field at ϕ=2° steps.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to *rho*.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.3.2 Dipole in Front of a Plate with HOBF

Create the dipole and the rectangular plate. Solve the model using the MoM with higher order basis functions (HOBF).

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Dipole in Front of a Plate with MoM and rename the file.
2. Activate higher order basis functions (HOBF) for the model.

> **Tip:** Open the **Solver settings** dialog and click the **General** tab. Select the **Solve MoM with higher order basis functions (HOBF)** check box. From the **Element order** list, select **Auto (default)**.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Dipole in Front of a Plate with MoM.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for Dipole in Front of a Plate with MoM.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.3.3 Dipole in Front of a Plate with FDTD

Simulate the dipole and rectangular plate using finite difference time domain (FDTD).

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Dipole in Front of a Plate with MoM and rename the file.
2. Activate the FDTD solver.

> **Tip:** Open the **Solver settings** dialog and click the **FDTD** tab. Select the **Activate the finite difference time domain (FDTD) solver** check box.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests from the Dipole in Front of a Plate with MoM model.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for Dipole in Front of a Plate with MoM.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.3.4 Dipole in Front of a Plate with UTD

Simulate the dipole with method of moments (MoM) and the rectangular plate using uniform theory of diffraction (UTD).

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Dipole in Front of a Plate with MoM and rename the file.
2. Set the solver method for the rectangular plate to use UTD.

> **Tip:** Open the **Modify Face** dialog and click the **Solution** tab. From the **Solve with special solution method** list, select **Uniform theory of diffraction (UTD)**.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Dipole in Front of a Plate with MoM.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for Dipole in Front of a Plate with MoM.

> **Note:** No mesh triangles are created. Feko applies the UTD solution to the plate surface.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.3.5 Dipole in Front of a Plate with RL-GO

Simulate the dipole with method of moments (MoM) and the rectangular plate using ray launching geometrical optics (RL-GO).

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Dipole in Front of a Plate with MoM and rename the file.
2. Set the solver method for the rectangular plate to use RL-GO.

> **Tip:** Open the **Modify Face** dialog and click the **Solution** tab. From the **Solve with special solution method** list, select **Ray launching - geometrical optics (RL-GO)**.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Dipole in Front of a Plate with MoM.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for Dipole in Front of a Plate with MoM.

> **Note:** Triangle sizes are determined by the geometrical shape and not the operating wavelength.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.3.6 Dipole in Front of a Plate with PO

Simulate the dipole with method of moments (MoM) and the rectangular plate using physical optics (PO).

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Dipole in Front of a Plate with MoM and rename the file.
2. Set the solver method for the rectangular plate to use **Physical optics (PO) - always illuminated**.

> **Tip:** Open the **Modify Face** dialog and click the **Solution** tab. From the **Solve with special solution method** list, select **Physical optics (PO) - always illuminated**.

> **Note:** Use the "always illuminated" option since there is no shadowing effect in the model. The "always illuminated" option avoids the ray tracing and accelerates the physical optics solution.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Dipole in Front of a Plate with MoM.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for Dipole in Front of a Plate with MoM.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.3.7 Dipole in Front of a Plate with LE-PO

Simulate the dipole with method of moments (MoM) and the rectangular plate using large element physical optics (LE-PO).

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Dipole in Front of a Plate with MoM and rename the file.
2. Set the solver method for the rectangular plate to use **Large element PO - always illuminated**.

> **Tip:** Open the **Modify Face** dialog and click the **Solution** tab. From the **Solve with special solution method** list, select **Large Element PO - always illuminated**.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Dipole in Front of a Plate with MoM.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Mesh size** equal to **Custom**
2. Set the **Triangle edge length** equal to $a/4$.

> **Tip:** A small distance separates the dipole and rectangular plate. Use a finer than **Standard** mesh setting for the LE-PO.

3. Set the **Wire segment length** equal to $h/10$.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.3.8 Viewing the Results

View and post-process the results in POSTFEKO.

1. Compare the gain (in dB) of the requested far field patterns on a polar plot.

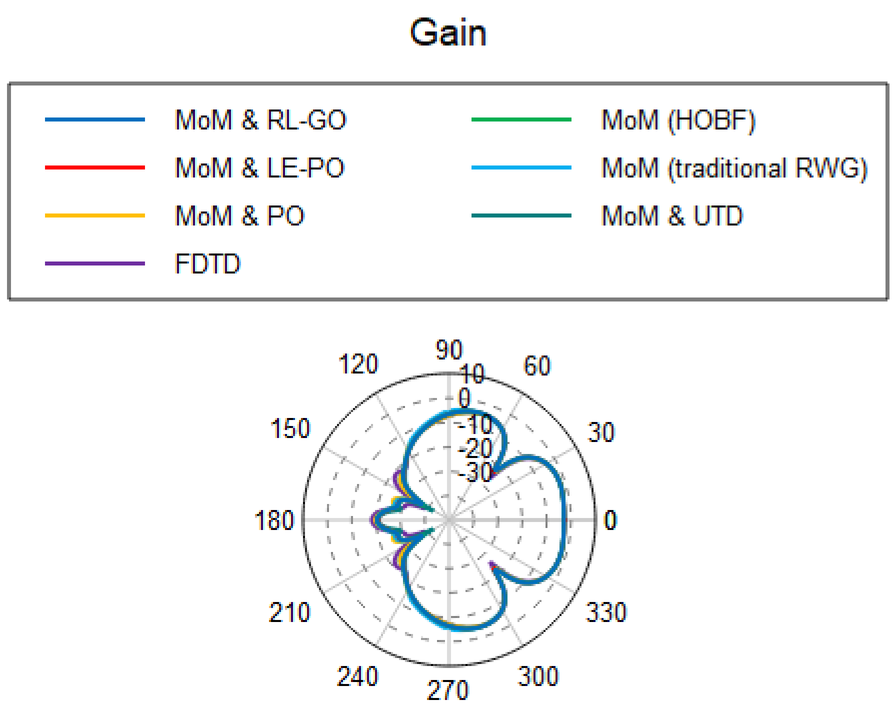


Figure 6: A polar plot of the requested radiation patterns for the different solvers in POSTFEKO.

2. View the runtime and memory requirements for each model in their respective `.out` files. Open the `.out` file in the output file viewer (in POSTFEKO), or in any other text file viewer.
3. Use the MoM solution as reference.

Table 1: Memory and runtime requirements for the different solvers normalised to the MoM solution.

Solution method	Memory (% of MoM)	Runtime (% of MoM)
HOBF	2.3	34
FDTD	31	161
UTD	0.05	0.5
RL-GO	11.5	3.4
PO	0.95	3.3

Solution method	Memory (% of MoM)	Runtime (% of MoM)
LE-PO	0.11	1.2

A.4 Monopole Antenna on a Finite Ground Plane

Calculate the radiation pattern for a wire monopole antenna on a finite ground plane. The ground plane is modelled as a circular PEC ground plane.

The circumference of the circular ground is 3 wavelengths at 75 MHz. The wire radius of the monopole antenna is 1 mm.

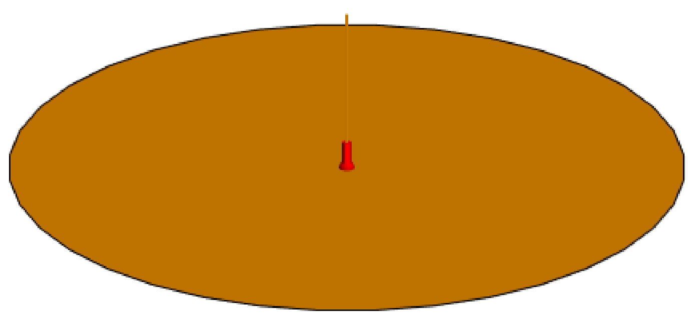


Figure 7: A 3D view of the monopole on a finite circular ground.

A.4.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
   - *freq* = 75e6 (The operating frequency.)
   - *lambda* = *c0*/*freq* (The wavelength in free space.)
   - *groundRadius* = 2 (Radius of the ground plane.)
   - *wireRadius* = 1e-3 (Radius of the wire.)
2. Create the circular ground.
   a) Create an ellipse.
      - **Centre point**: (0, 0, 0)
      - **Radius (U)**: *groundRadius*
      - **Radius (V)**: *groundRadius*
      - **Label**: `ground`

3. Create the monopole.
   a) Create a line.
      - **Start point**: (0, 0, 0)
      - **End point**: (0, 0, *lambda*/4)
      - **Label**: `monopole`
4. Union `ground` and `monopole`.
5. Add a wire port to the middle of the line.

   > **Tip:** Use the port preview to ensure the port is located at the junction between the wire and the ground plane. If the port is not located at the junction, change the port location to **Start**.

6. Add a voltage source to the port. (1 V, 0°, 50 Ω).
7. Set the frequency to *freq*.

A.4.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Create a full 3D far field request. Sample the far field at θ=2° and ϕ=2° steps.
2. Create a currents request (all currents).

A.4.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to *wireRadius*.

A.4.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.4.5 Viewing the Results

View and post-process the results in POSTFEKO.

1. View the total gain (in dB) in a vertical cut of the requested far field pattern using a polar plot.

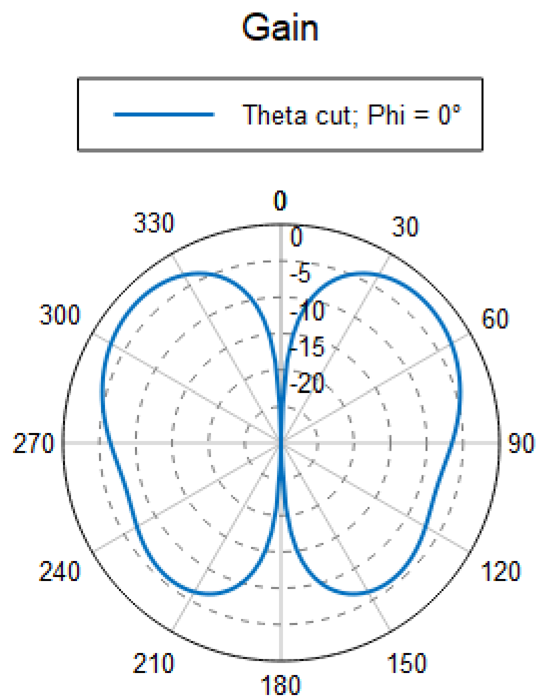


Figure 8: A polar plot of the far field gain (dB) in a vertical cut at 75 MHz viewed in POSTFEKO.

2. View the 3D gain pattern in the 3D view in POSTFEKO.

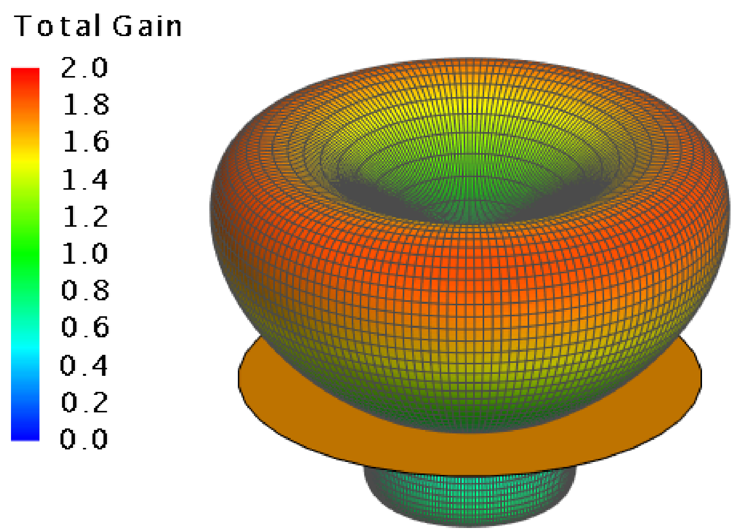


Figure 9: A 3D plot of the antenna gain in POSTFEKO.

Hide the far fields in the 3D view.

3. View the currents on the finite ground plane.

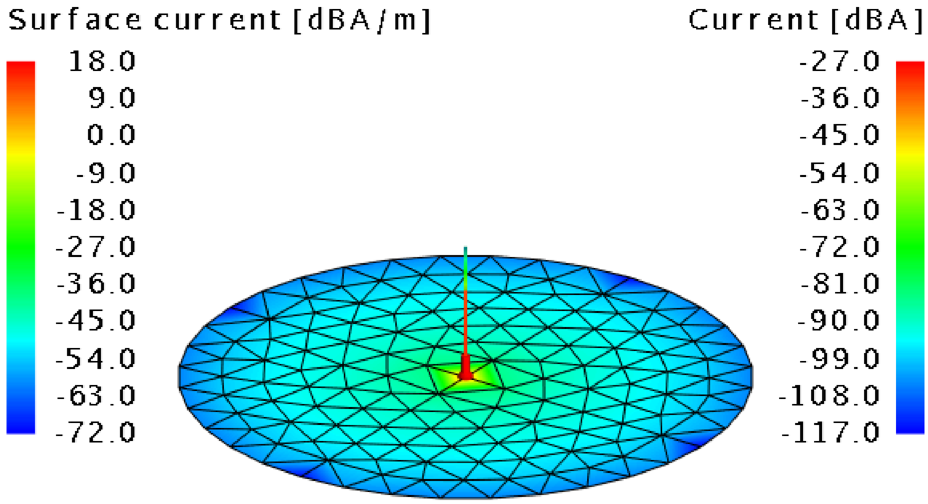


Figure 10: 3D view of the currents on the ground plane in POSTFEKO.

4. View the phase variation of the currents using animation.

> **Tip:** On the **Animate** tab, in the **Control** group, click the **Play** icon.

A.5 Yagi-Uda Antenna Above a Real Ground

Calculate the radiation pattern for a horizontally polarised Yagi-Uda antenna consisting of a dipole, a reflector and three directors at 400 MHz. The antenna is located 3 m above a real ground which is modelled with the Green's function formulation.

Figure 11: A 3D view of the Yagi-Uda antenna suspended over a real ground.

A.5.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
    - *freq* = 400e6 (The operating frequency.)
    - *lambda* = *c0*/*freq* (The wavelength in free space.)
    - *lr* = 0.477**lambda* (Length of the reflector.)
    - *li* = 0.451**lambda* (Length of the active element.)
    - *ld* = 0.442**lambda* (Length of the directors.)
    - *d* = 0.25**lambda* (Spacing between elements.)
    - *h* = 3 (Height of the antenna above ground.)
    - *epsr* = 10 (Relative permittivity of the ground.)
    - *sigma* = 1e-3 (Conductivity of the ground.)
    - *wireRadius* = 1e-3 (Radius of the wire.)

2. Create the dipole (driven element) of the Yagi-Uda antenna.
   a) Create a line.
      - **Start point**: (0, - *li*/2, *h*)
      - **End point**: (0, *li*/2, *h*)
      - **Label**: `activeElement`
   b) Add a wire port to the middle of the line.
   c) Add a voltage source to the port. (1 V, 0°, 50 Ω).
3. Create the reflector of the Yagi-Uda antenna.
   a) Create a line.
      - **Start point**: (-*d*, -*lr*/2, *h*)
      - **End point**: (-*d*, *lr*/2, *h*)
      - **Label**: `reflector`
4. Create the three directors of the Yagi-Uda antenna.
   a) Create three lines.

Line	Start point	End point	Label
1	(*d*, -*ld*/2, *h*)	(*d*, *ld*/2, *h*)	`director1`
2	(2**d*, -*ld*/2, *h*)	(2**d*, *ld*/2, *h*)	`director2`
3	(3**d*, -*ld*/2, *h*)	(3**d*, *ld*/2, *h*)	`director3`

5. Create a new dielectric called `ground` with relative permittivity set to *epsr* and conductivity to *sigma*.
6. Set the lower half space to **ground** using the exact Sommerfeld integrals.
7. Set the frequency to *freq*.

A.5.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a vertical far field request (-90°≤θ≤90°, with ϕ=0°). Sample the far field at θ=0.5° steps.

a) Change the workplane origin to (0,0,3).

A.5.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to *wireRadius*.

A.5.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

> **Note:** The following warning may be encountered when running the Solver:
>
> ```
> Directivity cannot be computed for far field calculations involving the
> planar multilayer Green's function with losses in the dielectric layers,
> gain will be computed instead.
> ```
>
> Losses cannot be calculated in an infinitely large medium as is required for the extraction of antenna directivity information.
>
> Avoid this warning by requesting the far field gain instead of the directivity. Open the **Request/Modify far fields** dialog, click the **Advanced** tab and then click **Gain**.

A.5.5 Viewing the Results

View and post-process the results in POSTFEKO.

1. View the gain (in dB) of the requested far field pattern on a polar plot.
2. Compare the results to a similar model where the ground plane is removed.

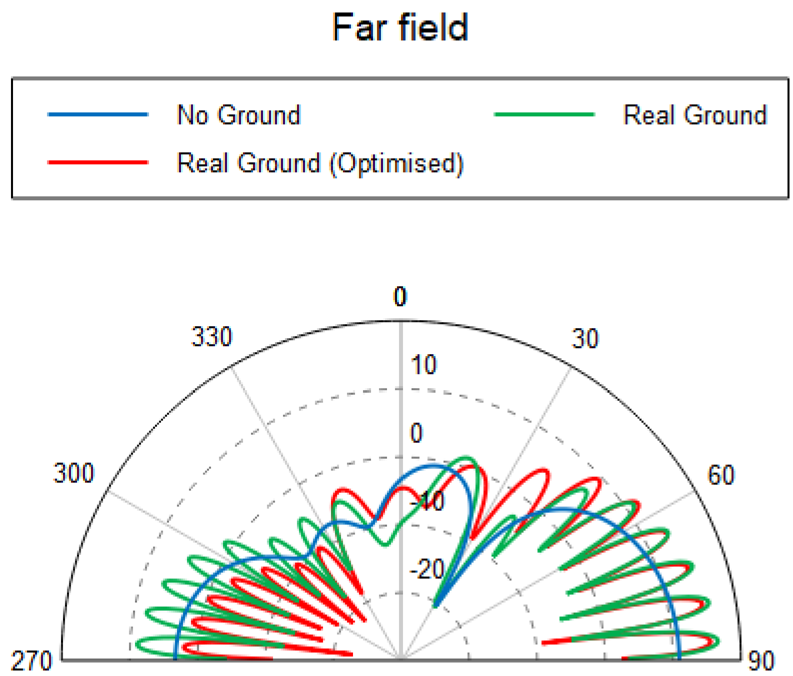


Figure 12: The gain pattern (in dB) of the Yagi-Uda antenna with no ground, a real ground, and the optimised pattern with a real ground.

Note: Observe the effect of the ground plane on the radiation pattern.

A.6 Pattern Optimisation of a Yagi-Uda Antenna

Optimise a Yagi-Uda antenna design to achieve a specific radiation pattern and gain at 1 GHz. The Yagi-Uda antenna consists of a dipole, reflector and two directors.

The initial antenna design is created from basic formulae. The antenna design is then optimised to obtain a gain above 8 dB in the main lobe (-30°≤ϕ≤30°) and below -7 dB in the back lobe (90°≤ϕ≤270°).

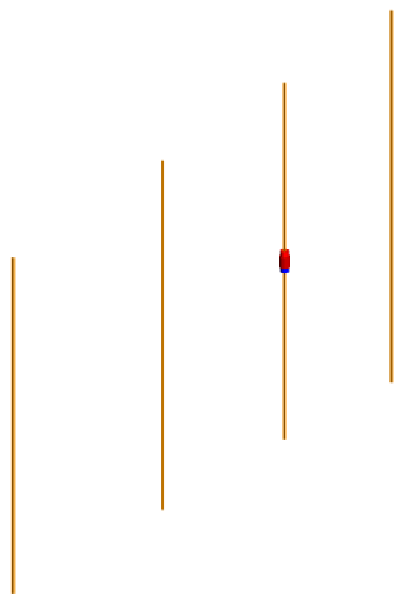


Figure 13: A 3D view of the Yagi-Uda antenna.

A.6.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
   - *freq* = 1e9 (The operating frequency.)
   - *lambda* = *c0*/*freq* (The wavelength in free space.)
   - *L0* = 0.2375 (Length of the reflector element in wavelengths.)
   - *L1* = 0.2265 (Length of the driven element in wavelengths.)
   - *L2* = 0.2230 (Length of the first director in wavelengths.)
   - *L3* = 0.2230 (Length of the second director in wavelengths.)
   - *S0* = 0.3 (Distance between the reflector and driven element in wavelengths.)
   - *S1* = 0.3 (Distance between the driven element and the first director in wavelengths.)
   - *S2* = 0.3 (Distance between the two directors in wavelengths.)

- *r* = 1e-4 (Wire radius.)

2. Set the incident power for the 50 Ω transmission line to 1 W.
3. Create the dipole (driven element) in the Yagi-Uda antenna.
   a) Create a line.
      - **Start point**: (0, 0, *-L1*lambda*)
      - **End point**: (0, 0, *L1*lambda*)
      - **Label**: `activeElement`

   b) Add a wire port (vertex) to the middle of the line.
   c) Add a voltage source to the port. (1 V, 0°, 50 Ω).
4. Create the reflector in the Yagi-Uda antenna.
   a) Create a line.
      - **Start point**: (*-S0*lambda*, 0, *-L0*lambda*)
      - **End point**: (*-S0*lambda*, 0, *L0**lambda)
      - **Label**: `reflector`
5. Create the first director in the Yagi-Uda antenna.
   a) Create a line.
      - **Start point**: (*S1*lambda*, 0, *-L2*lambda*)
      - **End point**: (*S1*lambda*, 0, *L2**lambda)
      - **Label**: `director1`
6. Create the second director in the Yagi-Uda antenna.
   a) Create a line.
      - **Start point**: ((*S1+ S2*)*lambda*, 0, *-L3*lambda*)
      - **End point**: ((*S1+S2*)*lambda*, 0, *L3*lambda*)
      - **Label**: `director2`
7. Set the frequency to `freq`.

A.6.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a horizontal far field request (0°≤ϕ≤180°, with θ=90°). Sample the far field at ϕ=2° steps.

a) Rename the label to `H_plane`.

A.6.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Mesh size** equal to **Standard**.
2. Set the **Wire segment radius** equal to *r*.

3. Mesh the model.

A.6.4 Adding an Optimisation Search

Add the optimisation search in CADFEKO.

1. Add an optimisation search. Use the **Simplex (Nelder-Mead)** method and **Low** accuracy.
2. Specify the optimisation parameters.
   a) On the **Optimisation parameters** dialog, on the **Variables** tab, define the following variables:

Variable	Min value	Max value	Start value	Grid points
L0	0.15	0.35	0.2375	Empty
L1	0.15	0.35	0.2265	Empty
L2	0.15	0.35	0.22	Empty
L3	0.15	0.35	0.22	Empty
S0	0.1	0.32	0.3	Empty
S1	0.1	0.32	0.3	Empty
S2	0.1	0.32	0.3	Empty

   b) On the **Constraints** tab, define the constraints. The reflector element is required to have a greater length than the director elements.
      - *L2 < L0*
      - *L3 < L0*
3. Create optimisation mask 1 to define the upper boundary for the gain (in dB).
   1. For 0° to 88°: gain < 15 dB
      - The value of 15 dB in the forward direction is selected knowing that this antenna will not be able to achieve 15 dB gain. It will not effect the optimisation.
   2. For 90° to 180°: gain < -7 dB
      - The value of -7 dB will have an effect on the optimisation and determines the size of the back lobes that we are willing to accept.
      - **Label**: Mask_max
4. Create optimisation mask 2 to define the lower boundary for the gain (in dB).
   1. For 0° to 30°: gain > 8 dB
      - The value of 8 dB is selected as the minimum desired main lobe gain.
   2. For 32° to 180°: gain > -40 dB

- The value of -40 dB outside the main lobe is selected arbitrarily low and will not affect the optimisation.
- **Label**: Mask_min

5. Define two far field goals. Use the **H_plane** far field request to optimise the vertically polarised gain in dB (10*log[]). The weighting for both goals are equal since both goals are of equal importance.
   - Set the 1st goal to be greater than **Mask_min**.
   - Set the 2nd goal to be less than **Mask_max**.

A.6.5 Running the Optimisation

Run OPTFEKO to optimise the model according to requirements. During optimisation, OPTFEKO will call the Solver as required.

1. Run OPTFEKO.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun OPTFEKO (if applicable).

A.6.6 Viewing the Results

View and post-process the results in POSTFEKO.

1. Compare the radiation pattern of the antenna for both the initial and the optimised antenna design. The gain in the back-lobe region (between 90° and 180°) is reduced to -7 dB. The gain in the main-lobe region (between 0° and 30°) is above 8 dB.

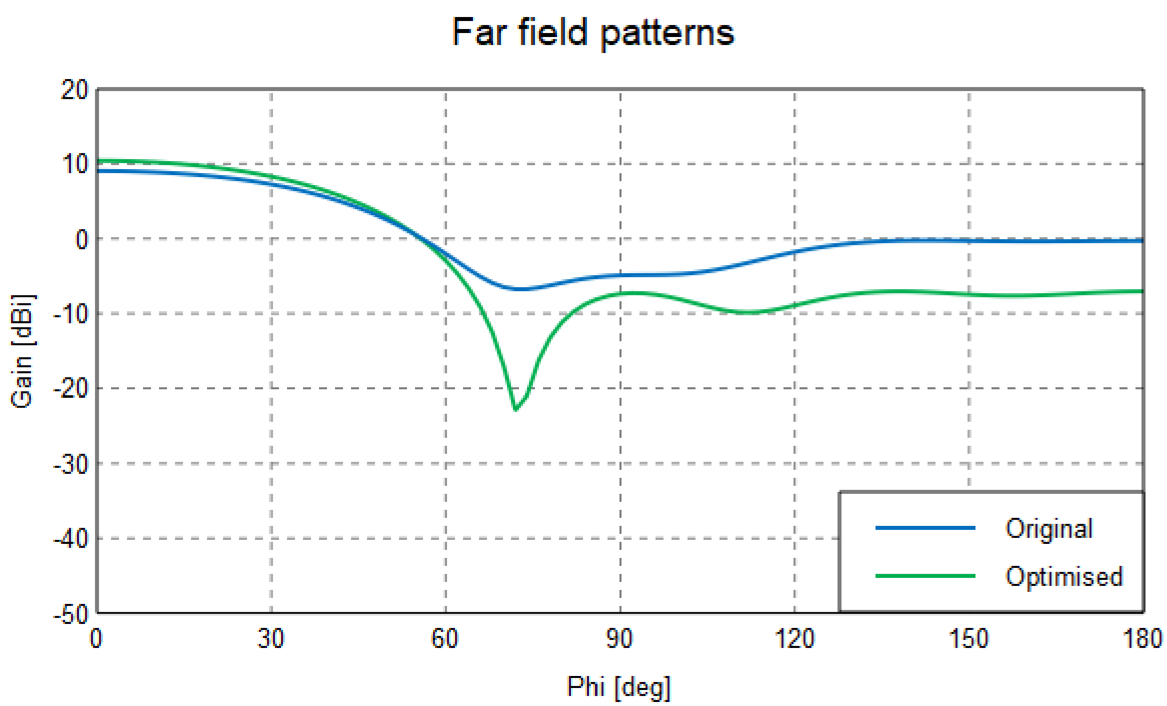


Figure 14: The vertically polarised gain of the Yagi-Uda antenna before and after optimisation.

> **Tip:** To view the masks on the same graph, drag each mask from the **model browser** onto the graph, then apply a scale of 180 (**Transform axis**) to each mask's trace.

2. View the optimum parameter values found during the optimisation search in the optimisation log file.

```
Value of the aim function (analysis no. 181) is   4.954552265e+02
Optimum found for these parameters:
l0 = 2.464426480e-01
l1 = 2.318887084e-01
l2 = 2.315246187e-01
l3 = 2.214966779e-01
s0 = 2.280809962e-01
s1 = 2.488307711e-01
s2 = 2.975472065e-01
No. of the last analysis: 212
```

A.7 Log Periodic Dipole Array Antenna

Calculate the radiation pattern and input impedance for a log periodic dipole array (LPDA) antenna. Non-radiating transmission lines are used to model the boom of the LPDA antenna.

The antenna operates at 46.29 MHz, with a wide operational bandwidth stretching from 35 MHz to 60 MHz.

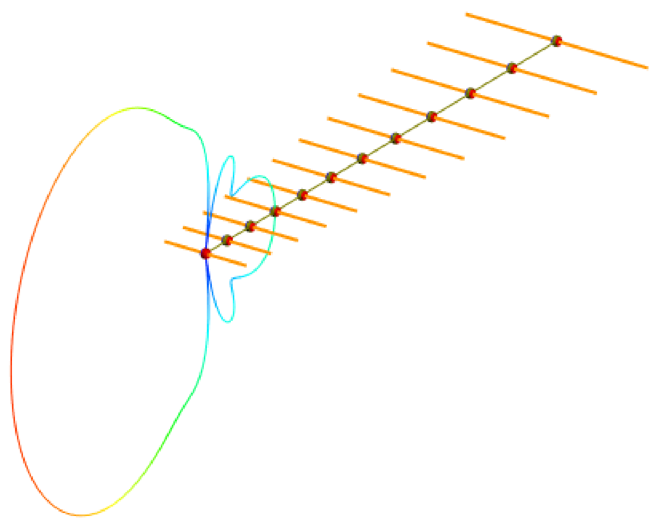


Figure 15: 3D view of the LPDA antenna with current distribution and far fields.

A.7.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
    - *freq* = 46.29e6 (The operating frequency.)
    - *lambda* = *c0*/*freq* (The wavelength in free space.)
    - *tau* = 0.93 (The growth factor.)
    - *sigma0* = 0.7 (Spacing)
    - *sigmaN* = *sigma(N-1)*/*tau*, where *N* is iterated from 1 to 11 with an increment of 1.
    - *d0* = 0 (Position of the first element.)
    - *dN* = *d(N-1)* – *sigmaN*, where *N* is iterated from 1 to 11 with an increment of 1.
    - *len0* = 2 (Length of the first element.)
    - *lenN* = *len(N-1)*/*tau*, where *N* is iterated from 1 to 11 with an increment of 1.
    - *rad0* = 0.00667 (Radius of the first element.)

- *radN* = *rad(N-1)/tau*, where *N* is iterated from 1 to 11 with an increment of 1.
- *Zline* = 50 (Transmission line impedance.)
- *Zload* = 50 (Shunt load resistance.)

2. Create twelve dipoles.
   a) Create lines 0 to *N*, where *N* is iterated from 0 to 11 with an increment of 1.
      - **Start point**: (*dN*, -*lenN*/2, 0)
      - **End point**: (*dN*, *lenN*/2, 0)
   b) Add a wire port to the middle of the line.
   c) Number the ports from 0 to 11.

3. Add a voltage source to the first dipole[1] (1 V, 0°, 50 Ω).

4. Create eleven transmission lines to connect the dipoles.
   a) Create transmission lines 1 to *N*, where *N* is iterated from 1 to 11 with an increment of 1.
      - **Definition method**: **Z0, length, attenuation**
      - **Transmission line length**: `sigmaN`
      - **Real part of Z0 (Ohm)**: `Zline`
      - **Imaginary part of Z0 (Ohm)**: `0`
      - **Attenuation (dB/m)**: `0`
      - Select the **Cross input and output ports** check box to allow the correct transmission line orientation.

5. Connect *TransmissionLineN* between *Port(N-1)* and *Port(N)* in the **Network schematic** view, where *N* is iterated from 1 to 11 with an increment of 1.

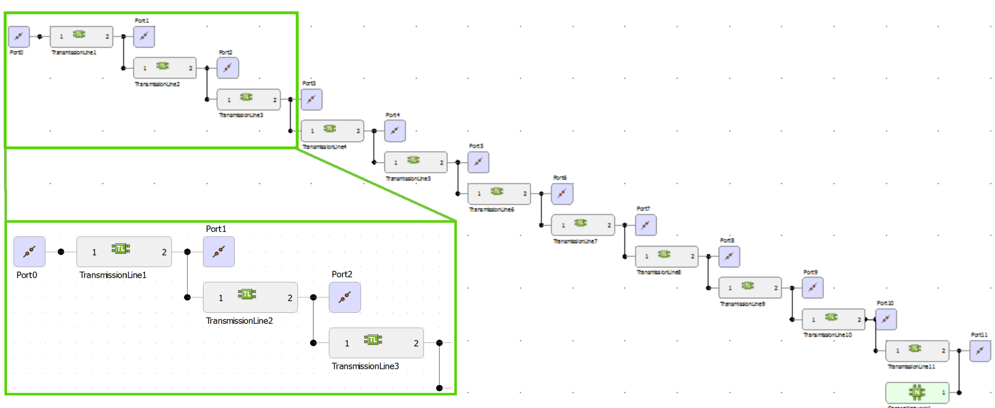


Figure 16: The network schematic view showing the connected transmission lines, general networks and ports.

6. Define a shunt load using a general network.
   - Specify the one-port Y-matrix manually.
   - Y_{11} = 1/*Zload*

1. This is *Line0* with *WirePort0*.

7. View transmission line 11 in the **Network schematic** view.
    a) Connect the general network to port 11.
8. Set the continuous frequency range from 35 MHz to 60 MHz.

A.7.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Create a vertical far field request (-180°≤θ≤180°, with ϕ=0°). Sample the far field at θ=2° steps.
2. Create a horizontal far field request (0°≤ϕ≤360°, with θ=90°). Sample the far field at ϕ=2° steps.

A.7.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Local wire radius** for each of the twelve dipoles to `radN`.

    > **Tip:** Open the **Modify Edge** dialog, and on the **Properties** tab and select the **Local wire radius** check box.

2. Set the **Mesh size** equal to **Standard**.
3. Set the **Wire segment radius** equal to `0.01`.

A.7.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.7.5 Viewing the Results

View and post-process the results in POSTFEKO.

1. View the vertical gain (in dB) at 46.29 MHz of the requested far field pattern on a polar plot.

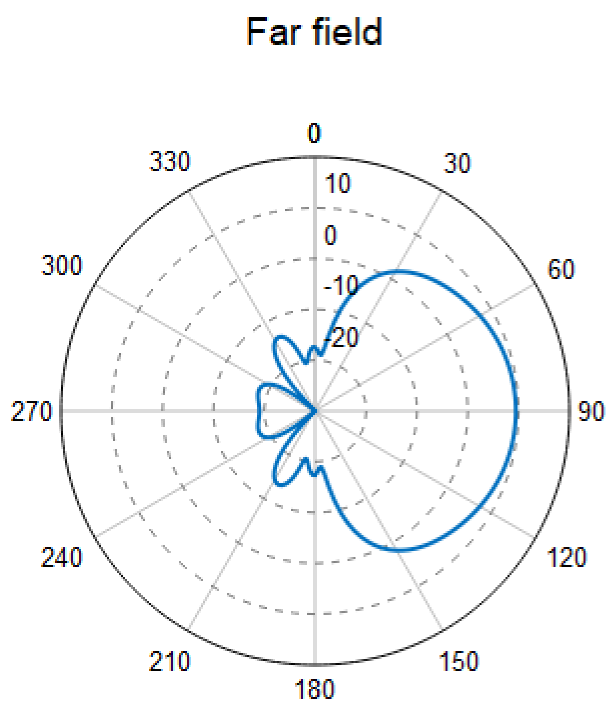


Figure 17: The vertical gain of the LPDA antenna at 46.29 MHz.

> **Tip:** If the exact frequency is not available in the drop-down list, select **Frequency in range** and enter the value for the frequency in Hz.

2. View the input impedance (real and imaginary) over the operating band on a Cartesian graph.

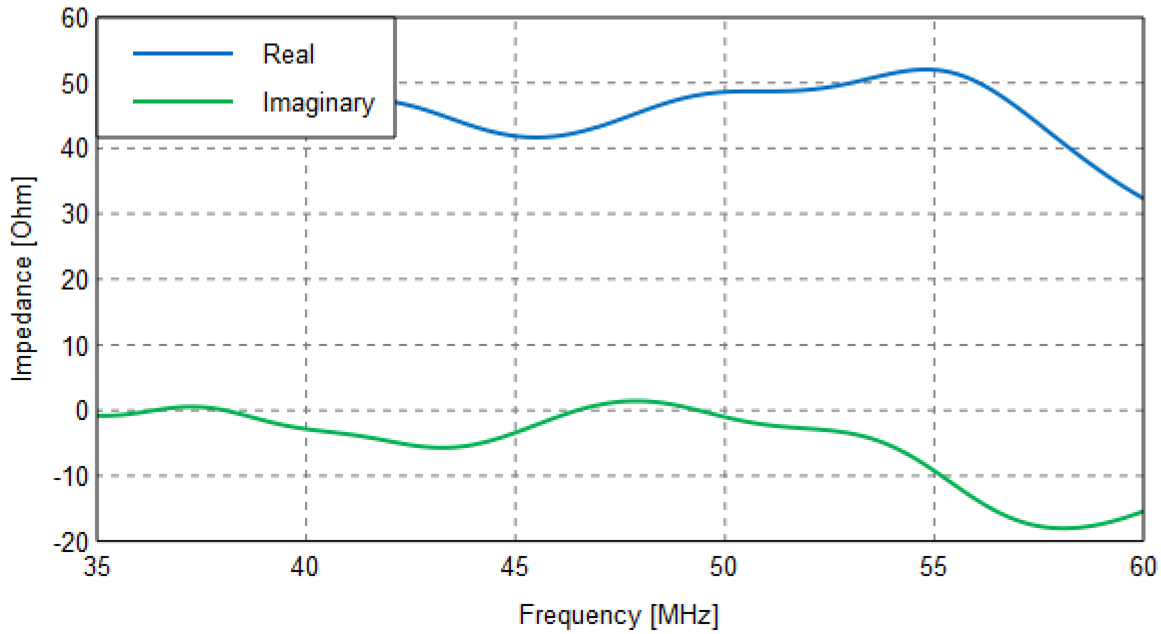


Figure 18: The input impedance (real and imaginary) of the LPDA antenna over the operating band.

A.8 Microstrip Patch Antenna

Model a microstrip patch antenna using two feed methods (pin feed, microstrip edge feed). The dielectric substrate is considered as a finite substrate and an infinite planar multilayer substrate.

The far field results are compared for the following configurations:

- Pin-fed, finite ground and solved using MoM (SEP).
- Pin-fed, finite ground and solved using the FDTD.
- Pin-fed, infinite substrate and solved using the MoM with the planar Green's function for multilayered media.
- Edge-fed, infinite substrate and solved using the MoM with the planar Green's function for multilayered media.

The simulation time and resource requirements are greatly reduced when using an infinite plane, although the model is a less accurate representation of the physical antenna.

> **Tip:** Each model uses its predecessor as a starting point. Create the models in their presentation order. Save each model to a new location to keep them.

A.8.1 Pin-Fed, SEP Model

Model a microstrip patch antenna using a feed pin and a finite substrate. The patch antenna is solved with the MoM (SEP).

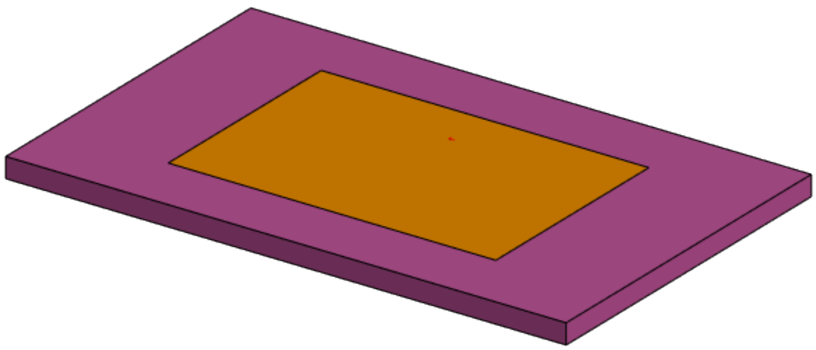


Figure 19: A 3D view of the pin-fed microstrip patch antenna on a finite ground.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to millimetres.
2. Define the following variables:
   - *epsr* = 2.2 (The relative permittivity of the substrate.)
   - *freq* = 3e9 (The centre frequency.)
   - *lambda* = *c0*/*freq**1e3 (The wavelength in free space.)
   - *lengthX* = 31.1807 (The width of the patch in the X direction.)
   - *lengthY* = 46.7480 (The depth of the patch in the Y direction.)
   - *offsetX* = 8.9 (The location of the feed.)
   - *substrateLengthX* = 50 (The width of the substrate in the X direction.)
   - *substrateLengthY* = 80 (The depth of the substrate in the Y direction.)
   - *substrateHeight* = 2.87 (The height of the substrate.)
   - *fmin* = 2.7e9 (The minimum frequency.)
   - *fmax* = 3.3e9 (The maximum frequency.)
   - *feedlineWidth* = 4.5 (The feedline width for the microstrip model feed.)
3. Create the patch.
   a) Create a rectangle.

- **Definition method: Base centre, width, depth**
- **Width**: *lengthX*
- **Depth** : *lengthY*
- **Label**: `patch`

4. Create the substrate.
   a) Create a cuboid.
   - **Definition method: Base corner, width, depth, height**
   - **Base corner**: (-*substrateLengthX*/2, -*substrateLengthY*/2, -*substrateHeight*)
   - **Width**: *substrateLengthX*
   - **Depth**: *substrateLengthY*
   - **Height**: *substrateHeight*
   - **Label**: `substrate`
5. Create the feed pin as a wire between the patch and the bottom of the substrate. Position the feed pin with an offset from the edge of the patch.
   a) Create a line.
   - **Start point**: (-*offsetX*, 0, -*substrateHeight*)
   - **End point**: (-*offsetX*, 0, 0)
6. Add a wire port to the start of the line.
7. Add a voltage source to the port. (1 V, 0°, 50 Ω).
8. Union all parts and rename the union to `antenna`.
9. Create a new dielectric medium with label `substrate` and with relative permittivity set to *epsr*.
10. Set the region of the cuboid to `substrate`.
11. Set the face of the patch and the bottom face of the substrate to PEC.
12. Set a continuous frequency range from *fmin* to *fmax*.
13. Specify the symmetry about the Y=0 plane as **Magnetic symmetry**.

> **Tip:** Exploit model symmetries (if it exists) in a large or complex model to reduce computational costs.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Create a vertical far field request (-90°≤θ≤90°, with ϕ=0°). Sample the far field at θ=2° steps.
2. Create a vertical far field request (-90°≤θ≤90°, with ϕ=90°). Sample the far field at θ=2° steps.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to `0.25`.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.8.2 Pin-Fed, FDTD Model

Model a microstrip patch antenna using a feed pin and a finite substrate. The patch antenna is solved with the finite difference time domain (FDTD).

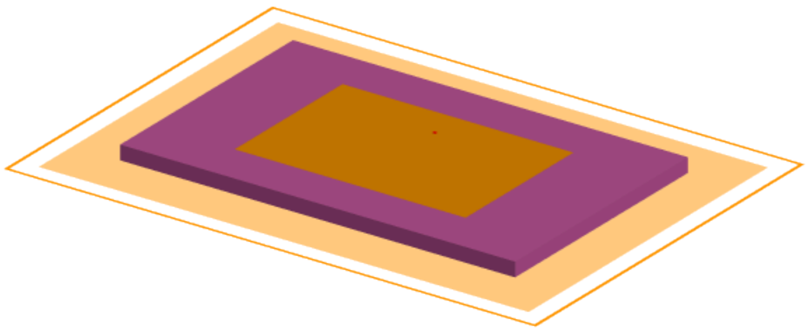


Figure 20: A 3D view of the pin-fed microstrip patch antenna on a finite ground.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Pin-Fed, SEP Model and rename the file.
2. Activate the FDTD solver.

   > **Tip:** Open the **Solver settings** dialog and click the **FDTD** tab. Select the **Activate the finite difference time domain (FDTD) solver** check box.

3. Change the continuous frequency range to linearly spaced discrete points ranging from *fmin* to *fmax* with the number of frequencies set to `51`.
4. Define the FDTD boundary condition settings.
   a) **Top (+Z)**, **-Y**, **+Y**, **-X** and **+X** boundaries:
      - **Boundary definition**: **Open**
      - Select the **Automatically add a free space buffer** check box.

   b) **Bottom (-Z)** boundary:
      - **Boundary definition**: **Perfect electric conductor (PEC)**
      - Select the **Do not add a free space buffer** check box.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Pin-Fed, SEP Model.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire radius** equal to `0.25`.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.8.3 Pin-Fed, Planar Multilayer Substrate Model

Model a microstrip patch antenna using a feed pin and a planar multilayer substrate (Green's functions). The patch antenna is solved with MoM.

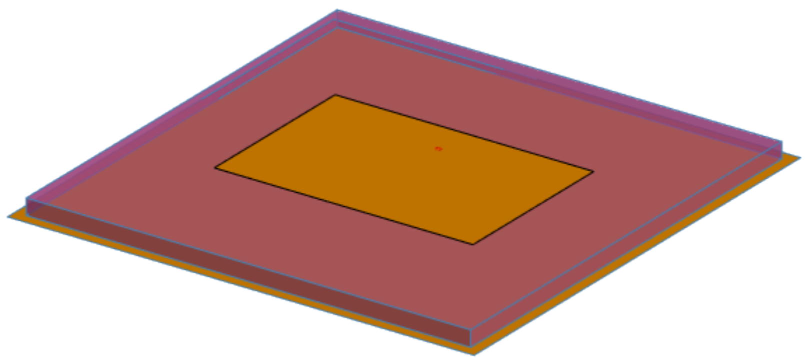


Figure 21: A 3D view of the pin-fed microstrip patch antenna on an infinite ground.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Pin-Fed, SEP Model and rename the file.
2. Delete the *substrate* part contained in the *antenna* part.
3. Add a planar multilayer substrate (infinite plane) with a conducting layer at the bottom.
    a) Add a **Plane / ground.**
        - Select **Planar multilayer substrate** from the drop-down list.
        - **Ground plane** (Layer 1): **PEC**
        - **Thickness** (Layer 1): *substrateHeight*
        - **Medium** (Layer 1): *substrate*

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Pin-Fed, SEP Model.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for Pin-Fed, SEP Model.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

> **Note:** The following warning may be encountered when running the Solver:
>
> ```
> Directivity cannot be computed for far field calculations involving the
> planar multilayer Green's function with losses in the dielectric layers,
> gain will be computed instead.
> ```
>
> Losses cannot be calculated in an infinitely large medium as is required for the extraction of antenna directivity information.
>
> Avoid this warning by requesting the far field gain instead of the directivity. Open the **Request/Modify far fields** dialog, click the **Advanced** tab and then click **Gain**.

A.8.4 Edge-Fed, Planar Multilayer Substrate Model

Model a microstrip patch antenna using an edge feed and a planar multilayer substrate (Green's functions). The patch antenna is solved with MoM.

> **Note:** This example is for demonstration purposes only. For practical applications the patch should be inset-fed by the feed line to improve the impedance match.
>
> To improve accuracy, the length of the edge used for the edge port should be less than 1/30 of a wavelength (around 3mm for this example). For demonstration purposes, an edge length of 4.5 mm is used.

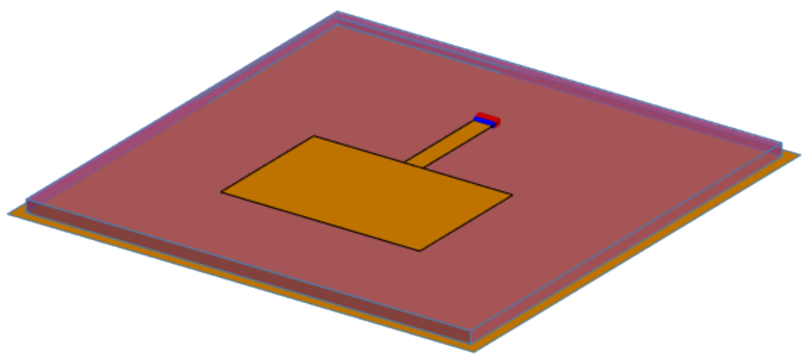


Figure 22: A 3D view of the edge-fed microstrip patch antenna on an infinite ground.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Pin-Fed, Planar Multilayer Substrate Model and rename the file.
2. Copy (duplicate) the *patch* part contained in the *antenna* part.
3. Delete the *antenna* part.
   The model should now only contain the *patch* part.
4. Create the feedline.
   a) Create a rectangle.
      - **Definition method**: **Base corner, width, depth**
      - **Base corner**: (-*lengthX*/2, -*feedlineWidth*/2, 0)
      - **Width**: -*lambda*/4
      - **Depth**: *feedlineWidth*
      - **Label**: `feedline`

5. Union all the parts.
6. Add a microstrip port at the edge of the feed line.
7. Add a voltage source to the port. (1 V, 0°, 50 Ω).

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Pin-Fed, SEP Model.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for Pin-Fed, SEP Model.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

> **Note:** The following warning may be encountered when running the Solver:
>
> ```
> Directivity cannot be computed for far field calculations involving the
> planar multilayer Green's function with losses in the dielectric layers,
> gain will be computed instead.
> ```
>
> Losses cannot be calculated in an infinitely large medium as is required for the extraction of antenna directivity information.
>
> Avoid this warning by requesting the far field gain instead of the directivity. Open the **Request/Modify far fields** dialog, click the **Advanced** tab and then click **Gain**.

A.8.5 Viewing the Results

View and post-process the results in POSTFEKO.

Compare the gain (in dB) of the requested far field patterns on a polar plot.

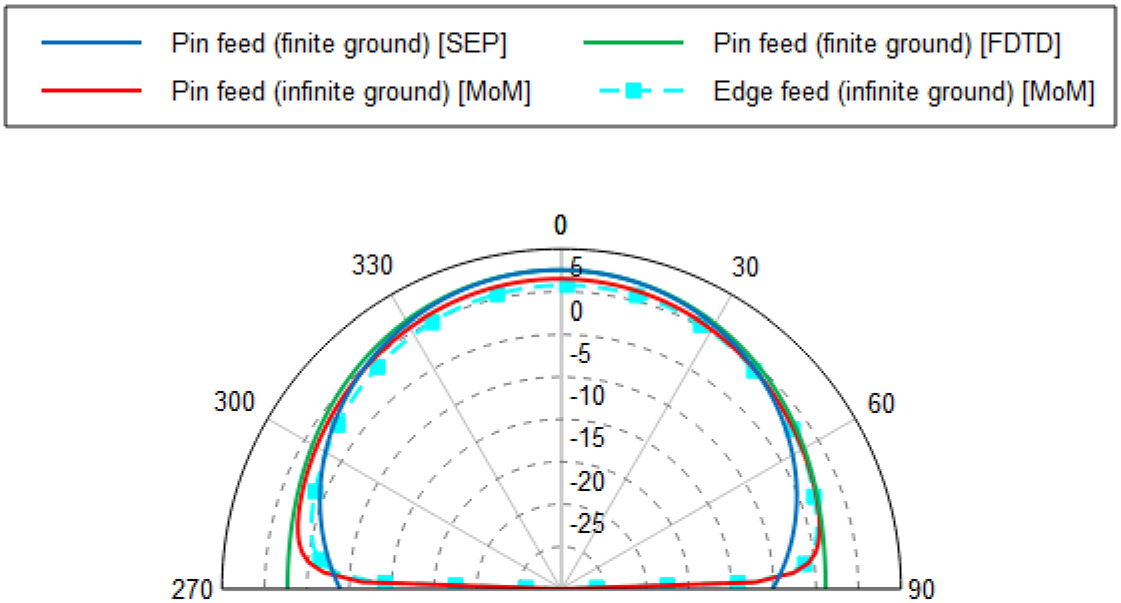


Figure 23: A polar plot of the requested E plane radiation pattern for the microstrip patch models in POSTFEKO.

> **Note:** Observe the impact of the different feeding mechanisms on the radiation pattern.

The best design for manufacturability (DFM) is the model using the finite ground. The finite ground version requires more time to solve compared to the infinite plane version.

A.9 Proximity Coupled Patch Antenna with Microstrip Feed

Calculate the input reflection coefficient of a proximity coupled patch antenna on an infinite substrate.

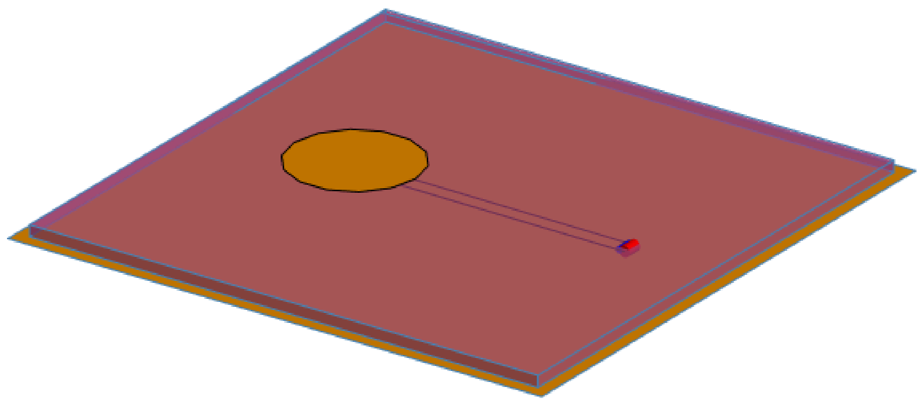


Figure 24: A 3D view of the proximity coupled microstrip fed patch.

A.9.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the **Model unit** to **Millimetres**.
2. Define the following variables.
   - *epsr* = `2.62` (The relative permittivity of the substrate.)
   - *patch_rad* = `17.5` (The radius of the patch.)
   - *line_len* = `79` (The length of the strip line.)
   - *line_width* = `4.373` (The width of the strip line.)
   - *offset* = `0` (The distance between the patch centre and the feed line.)
   - *substrate_d* = `3.18` (The height of the substrate.)
   - *f_min* = `2.8e9` (The minimum frequency.)
   - *f_max* = `3.2e9` (The maximum frequency.)
3. Create a dielectric medium.
   - **Relative permittivity**: *epsr*
   - **Dielectric loss tangent**: `0`

- **Label**: `substrate`

4. Create the circular patch.
   a) Create an ellipse.
      - **Centre point**: (0, 0, 0)
      - **Radius (U)**: *patch_rad*
      - **Radius (V)**: *patch_rad*
5. Create the feed line.
   a) Create a rectangle.
      - **Definition method**: **Base corner, width, depth.**
      - **Base corner**: (-*line_width*/2, 0, -*substrate_d*/2).
      - **Width (W)**: *line_width*
      - **Depth (D)**: *line_len*
6. Add a planar multilayer substrate (infinite plane) with a conducting layer at the bottom.
   a) Add a **Plane / ground.**
   - Select **Planar multilayer substrate** in the drop-down list.
   - **Ground plane** (Layer 1): *PEC*
   - **Thickness** (Layer 1): *substrate_d*
   - **Medium** (Layer 1): *substrate*
7. Create a microstrip port
   - Click on the short edge (beginning) of the feed line.

     > **Tip:** To select the edge, hide the **Plane / ground** or add a cutplane.
8. Add a voltage source to the port. (1 V, 0°, 50 Ω).
9. Set the frequency.
   - **Continuous (interpolated) range**
   - **Start frequency (Hz)**: *f_min*
   - **End frequency (Hz)**: *f_max*

A.9.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

No solution requests are required.

> **Note:** Input impedance results are always available for voltage sources.

A.9.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Adjust the sliders for the **Curved geometry approximation settings** on the **Advanced** tab. Create different meshes with these sliders and investigate the effect on the results.

A.9.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.9.5 Viewing the Results

View and post-process the results in POSTFEKO.

View the input reflection coefficient on a Smith chart.

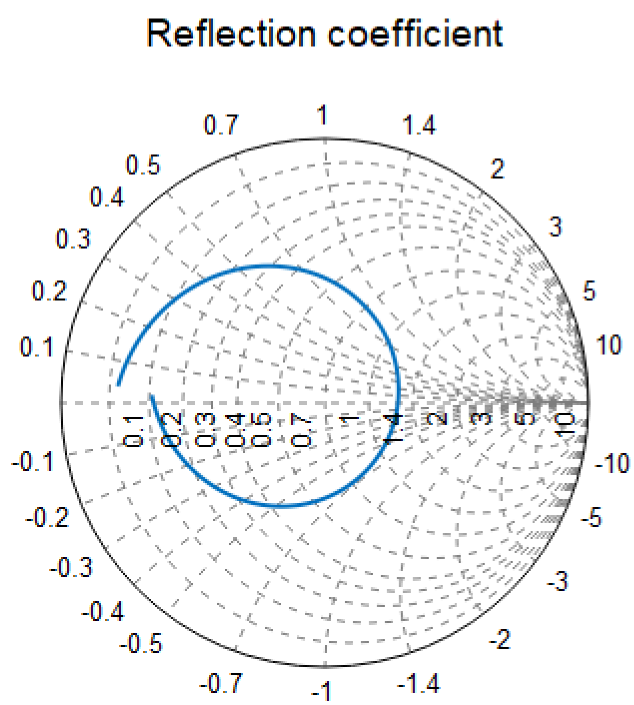


Figure 25: The input reflection coefficient of the proximity coupled patch in POSTFEKO.

A.10 Aperture Coupled Patch Antenna

Calculate the input reflection coefficient of an aperture coupled patch antenna. Use continuous frequency sampling to minimise runtime. Compare results for a finite and infinite dielectric.

Model the dielectric layers with two methods:

1. Model the patch antenna using a finite substrate where the aperture is modelled as an explicit (unmeshed) hole. The patch antenna is solved with the MoM (SEP).
2. Model the patch antenna using an infinite multilayer substrate where aperture triangles allow the energy to couple through an infinite PEC ground plane.

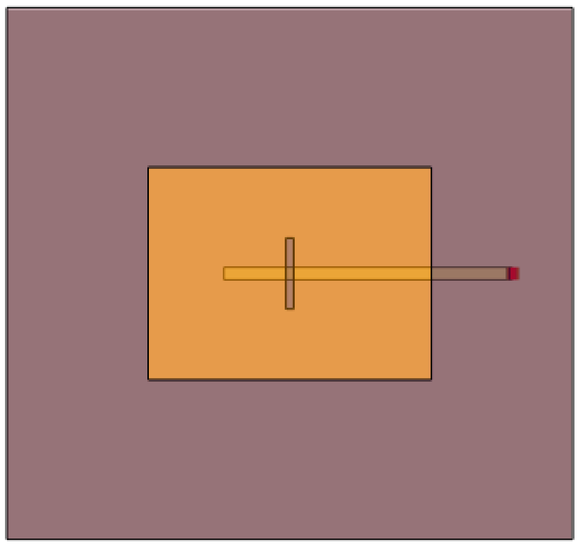


Figure 26: A 3D view of the aperture coupled patch antenna with microstrip feed.

A.10.1 SEP Model

Model the patch antenna using a finite substrate where the aperture is modelled as an explicit (unmeshed) hole. The patch antenna is solved with the MoM (SEP).

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to centimeters.
2. Define the following variables.
   - *epsr_a* = 10.2 (The relative permittivity of the bottom layer.)
   - *epsr_b* = 2.54 (The relative permittivity of the top layer.)
   - *f_min* = 2.1e9 (The minimum frequency.)
   - *f_max* = 2.3e9 (The maximum frequency.)
   - *lambda_a* = *c0*/*f_max*/sqrt(*epsr_a*)*100 (The wavelength in the bottom layer.)
   - *lambda_b* = *c0*/*f_max*/sqrt(*epsr_b*)*100 (The wavelength in the top layer.)
   - *d_a* = 0.16 (The height of the bottom layer.)
   - *d_b* = 0.16 (The height of the top layer.)
   - *patch_l* = 4.0 (The length of the patch antenna.)
   - *patch_w* = 3.0 (The width of the patch antenna.)
   - *grnd_l* = 2***patch_l* (The length of the substrate.)
   - *grnd_w* = 2.5***patch_w* (The width of the substrate.)
   - *feed_l* = *lambda_a* (The length of the microstrip feed line.)
   - *feed_w* = 0.173 (The width of the microstrip feed line.)
   - *stub_l* = 1.108 (Length of the matching stub on the microstrip feed line.)
   - *ap_l* = 1.0 (The length of the aperture.)
   - *ap_w* = 0.11 (The width of the aperture.)
3. Create a dielectric medium for the bottom layer.
   - **Relative permittivity**: *epsr_a*
   - **Dielectric loss tangent**: 0
   - **Label**: `bottom_layer`
4. Create a dielectric medium for the top layer.
   - **Relative permittivity**: *epsr_b*
   - **Dielectric loss tangent**: 0
   - **Label**: `top_layer`
5. Create the aperture.
   a) Create a rectangle.
      - **Definition method**: **Base centre, width, depth.**
      - **Base centre (C)**: (0, 0, 0)

- **Width (W)**: *ap_l*
- **Depth (D)**: *ap_w*
- **Label**: `aperture`

6. Create the finite ground plane.
   a) Create a rectangle.
      - **Definition method**: **Base centre, width, depth**
      - **Base centre (C)**: (0, 0, 0)
      - **Width (W)**: *grnd_w*
      - **Depth (D)**: *grnd_l*
      - **Label**: `ground`
7. Create the aperture in the ground.
   a) Subtract *aperture* from *ground*.
   b) Rename *Subtract1* to `slotted_ground`.

   The finite ground plane now has a hole at the centre where the aperture plate was defined.
8. Create the patch.
   a) Create a rectangle.
      - **Definition method**: **Base centre, width, depth.**
      - **Base centre (C)**: (0, 0, *d_b*)
      - **Width (W)**: *patch_w*
      - **Depth (D)**: *patch_l*
      - **Label**: *patch*
9. Create the microstrip feed line.
   a) Create a rectangle.
      - **Definition method**: **Base corner, width, depth**
      - **Base corner (C)**: (-*feed_w*/2, -*feed_l*/2 + *stub_l*, -*d_a*)
      - **Width (W)**: *feed_w*
      - **Depth (D)**: *feed_l*
      - **Label**: *feed*

The source will be a voltage source placed on an edge port.

10. Create a plate (via) that connects the ground plane and feed line.
    a) Create a rectangle at the end of the feed line in the XZ plane.
       1. **Definition method**: **Base corner, width, depth**
       2. Choose **Custom workplane** and change the workplane origin to (-*feed_w*/2, *feed_l*/2 + *stub_l*, -*d_a*).
       3. Rotate the workplane by 90° around the U axis to create the rectangle in the XZ plane.
       4. **Base corner (C)**: (0, 0, 0)
       5. **Width (W)**: *feed_w*
       6. **Depth (D)**: *d_a*
       7. **Label**: `feedVia`

> **Tip:** Rotate the workplane by selecting **Custom workplane** and the right-click context menu.

A positive terminal and a negative terminal are required for the edge port.

11. Split `feedVia` in the UV plane at (0, 0, -*d_a*/2).

   a) Rename the two resulting parts to *port_bottom* and *port_top* respectively.

12. Union *port_bottom* and *port_top* and rename the resulting part to `conducting_elements`.

13. Set the properties of all the faces to PEC.

> **Note:** This step ensures that the faces will remain PEC after future union operations.

14. Create the bottom dielectric layer.

   a) Create a cuboid.

   - **Definition methods**: **Base centre, width, depth**
   - **Base centre (C)**: (0, 0, -*d_a*)
   - **Width (W)**: *grnd_w*
   - **Depth (D)**: *grnd_l*
   - **Height (H)**: *d_a*
   - **Label**: `bottom_layer`

15. Create the top dielectric layer.

   a) Create a cuboid.

   - **Base centre (C)**: (0, 0, 0)
   - **Width (W)**: *grnd_w*
   - **Depth (D)**: *grnd_l*
   - **Height**: *d_b*
   - **Label**: `top_layer`

16. Union all parts.

17. Set the bottom region to `bottom_layer`.

18. Set the top region to `top_layer`.

19. Add an edge port to the edge that splits the via connection in half.

- The negative face corresponds to the face attached to the ground plane.
- The positive face is the opposite face.

> **Tip:** The polarity of the port is not relevant for single port models.

20. Add a voltage source to the port. (1 V, 0°, 50 Ω).

21. Set a continuous frequency range from *f_min* to *f_max*.

22. Specify the symmetry about the X=0 plane as **Magnetic symmetry**.

> **Tip:** Exploit model symmetries (if it exists) in a large or complex model to reduce computational costs.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a full 3D far field request.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Specify local mesh refinements.

a) Set a local mesh size of *lambda_b*/40 on all four edges of the patch.
b) Set a local mesh size of *ap_w**0.7 on all four edges of the aperture.
c) Set a local mesh size of *feed_w*/4 on the face of the feed.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.10.2 Aperture Triangles in an Infinite Ground Plane

Model the patch antenna using an infinite multilayer substrate where aperture triangles allow the energy to couple through an infinite PEC ground plane.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

Define the variables and media.

1. Repeat Step 1 to Step 4 in SEP Model.

Create the aperture.

2. Repeat Step 5 in SEP Model.
3. Set the solver method for the aperture face to use **Planar Green's function aperture**.

Create the patch.

4. Repeat Step 8 in SEP Model.

Create the microstrip feed line and plate (via) that connects the infinite ground plane and feed line.

5. Repeat Step 9 and Step 10 in SEP Model.

Create the positive and negative terminals of the edge port.

6. Repeat Step 11 in SEP Model.
7. Create an infinite ground plane using a planar multilayer substrate with a conducting layer at the bottom.
    a) Select **Plane / ground**.
        - Select **Planar multilayer substrate** in the drop-down list.
        - **Thickness** (Layer 1): *d_b*
        - **Medium** (Layer 1): *top_layer*
        - **Ground plane** (Layer 1): *PEC*
        - **Thickness** (Layer 2): *d_a*
        - **Medium** (Layer 2): *bottom_layer*
        - **Ground plane** (Layer 1): *None*
        - **Z value at the top of layer 1**: *d_a*
8. Union all parts.

Add an edge port, voltage source, specify the frequency and define symmetry.

9. Repeat Step 19 to Step 22 to in SEP Model.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for SEP Model.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Use the same mesh settings as for SEP Model.
2. Set the **Mesh size growth rate** to **Slow**.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.10.3 Viewing the Results

View and post-process the results in POSTFEKO.

1. Compare the input reflection coefficient of the two methods on a Smith chart.

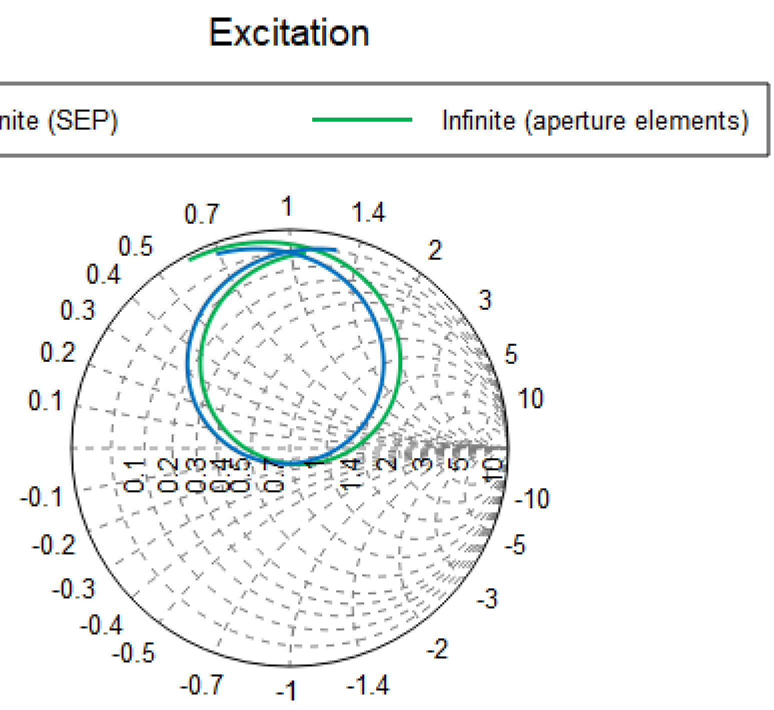


Figure 27: The input reflection coefficient of the aperture coupled patch in POSTFEKO.

Note: The model using an infinite plane is a good approximation of the SEP model.

2. Compare the realised gain (in dB) at boresight of both methods on a Cartesian graph.

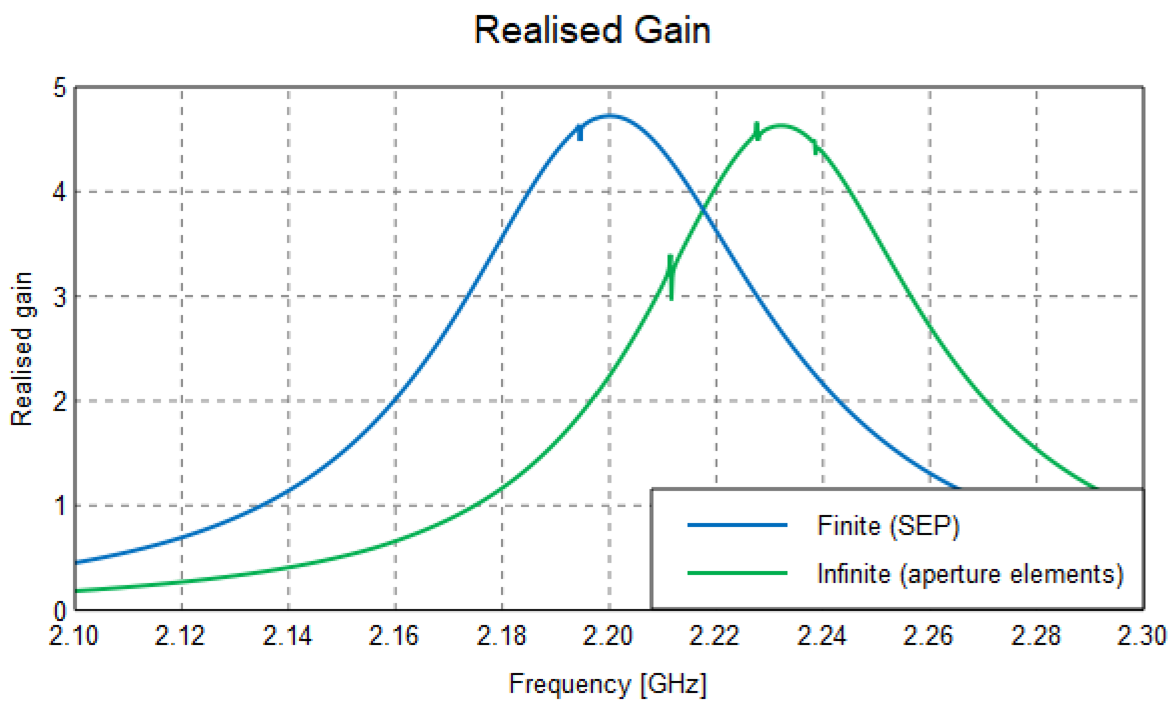


Figure 28: Far field realised gain over frequency.

Note: The far fields have a similar shape and the center frequency deviates by less than 2%. Increase the size of the finite substrate to obtain an even better comparison between the two methods.

3. Compare computational resources for the two methods.

Table 2: Memory and runtime requirements for the two methods.

Model	Approximate number of triangles	RAM [MByte]	Runtime [% of full SEP]
SEP	10000	3500	100
Infinite ground plane	1500	19	9.5

> **Note:** Use an infinite ground plane to reduce the number of triangles and the computational resources.

A.11 Different Ways to Feed a Horn Antenna

Calculate the far field pattern of a pyramidal horn antenna at 1.645 GHz.

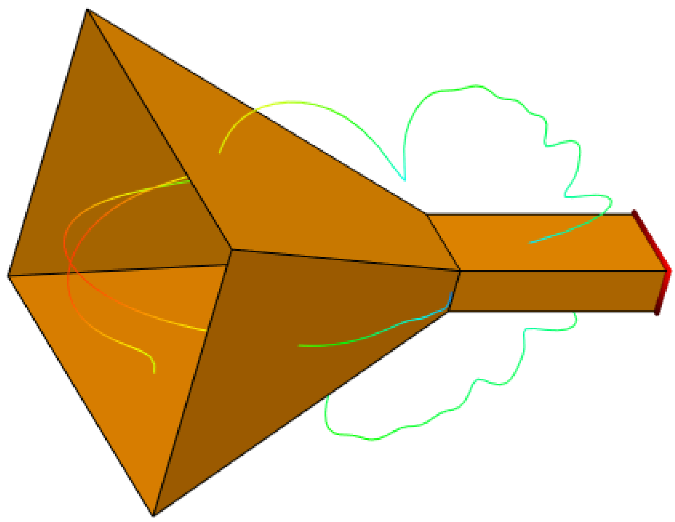


Figure 29: A 3D view of the pyramidal horn antenna with far field pattern cuts..

The far field results are compared for the following configurations:

1. Physical feed pin with voltage source.
2. Waveguide port with waveguide source.
3. FEM modal port with modal source using the finite element method (FEM).

A.11.1 Wire Pin Feed Model

Feed the horn antenna with a physical feed pin and voltage source.

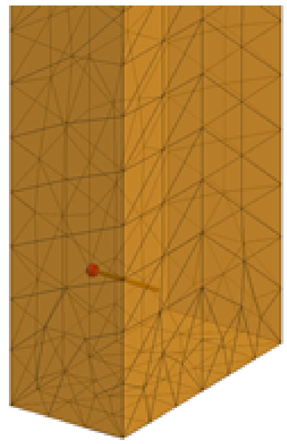


Figure 30: A 3D view of the horn with a wire pin feed.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to centimetres.
2. Define the following variables:
    - *freq* = 1.645e9 (The operating frequency.)
    - *lambda* = *c0*/*freq* * 100 (The wavelength in free space.)
    - *wa* = 12.96 (The width of the waveguide.)
    - *wb* = 6.48 (The height of the waveguide.)
    - *ha* = 55 (The width of the horn.)
    - *hb* = 42.8 (The height of the horn.)
    - *wl* = 30.2 (The length of the waveguide section.)
    - *fl* = `wl - lambda/4` (The position of the feed wire in the waveguide.)
    - *hl* = 46 (The length of the horn section.)
    - *pinlen* = *lambda* / 4.56 (The length of the pin.)
3. Create the waveguide section.
    a) Create a cuboid.
        - **Definition method**: **Base corner, width, depth, height**
        - **Base corner (C)**: (-*wa*/2, -*wb*/2, -*wl*)
        - **Width (W)**: *wa*
        - **Depth (D)**: *wb*

- **Height (H)**: *wl*

4. Delete the face coincident with the UV plane.
5. Create the horn section.
   a) Create a flare.
      - **Definition method**: **Base centre, width, depth, height, top width, top depth**
      - **Bottom width (Wb)**: *wa*
      - **Bottom depth (Db)**: *wb*
      - **Height (H)**: *hl*
      - **Top width (Wt)**: *ha*
      - **Top depth (Dt)**: *hb*
6. Delete the face at the origin and the face opposite to that.
7. Create the feed pin.
   a) Create a line.
      - **Start point**: (0, -*wb/2*, -*fl*)
      - **End point**: (0, -*wb/2* + *pinlen*, -*fl*)
8. Add a wire port (segment) to the base of the line.
9. Add a voltage source to the port. (1 V, 0°, 50 Ω).
10. Union all parts.
11. Set the frequency to *freq*.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a far field request for the E plane cut.

1. Create a vertical far field request (-180°≤θ≤180°, with ϕ=90°). Sample the far field at θ=2° steps.

Create a far field request for the H plane cut.

2. Create a horizontal far field request (-180°≤θ≤180°, with ϕ=0°). Sample the far field at θ=2° steps.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Mesh size** equal to **Coarse**.
2. Set the **Wire segment radius** equal to `0.1`.

> **Tip:** Keep the runtime at a minimum by using coarse meshing to reduce the number of triangles.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.11.2 Waveguide Feed Model

Feed the horn antenna with a waveguide port and waveguide source.

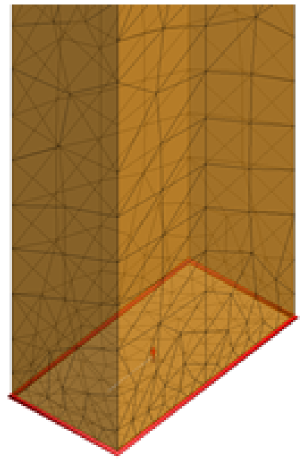


Figure 31: A 3D view of the horn with a waveguide port.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

The wire feed model is changed to use a waveguide feed.

1. Use the model considered in Wire Pin Feed Model and rename the file.
2. Delete the voltage source.
3. Delete the wire port.
4. Delete the line.
5. Apply a waveguide port to the back face of the horn.

   Note: The face type for the port (rectangular, coaxial or circular) is determined automatically by CADFEKO.

   Tip: Inspect the port to check if the propagation direction and orientation of the port is correct.

6. Add a waveguide source on the waveguide port using the default settings.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Wire Pin Feed Model.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the same mesh settings as for Wire Pin Feed Model.

   > **Note:** Removing the line (wire) from the model eliminates the need for the **Wire segment radius** specification.

2. Set a local mesh size of `lambda/20` on the back face of the waveguide.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.11.3 FEM Modal Port Feed Model

Feed the horn antenna with a FEM modal port and FEM modal source.

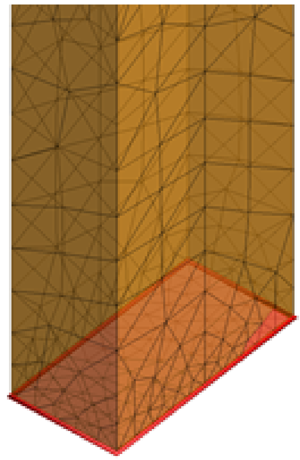


Figure 32: A 3D view of the horn with a FEM modal port with a FEM modal source.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to centimetres.
2. Define the following variables:
   - *freq* = 1.645e9 (The operating frequency.)
   - *lambda* = *c0*/*freq* * 100 (The wavelength in free space.)
   - *wa* = 12.96 (The width of the waveguide.)
   - *wb* = 6.48 (The height of the waveguide.)
   - *ha* = 55 (The width of the horn.)
   - *hb* = 42.8 (The height of the horn.)
   - *wl* = 30.2 (The length of the waveguide section.)
   - *fl* = `wl - lambda/4` (The position of the feed wire in the waveguide.)
   - *hl* = 46 (The length of the horn section.)
   - *pinlen* = *lambda* / 4.56 (The length of the pin.)
3. Define a dielectric medium, *air*.
   - **Relative permittivity**: `1`
   - **Dielectric loss tangent**: `0`
   - **Label**: `air`
4. Create the waveguide section.
   a) Create a cuboid.

- **Definition method**: **Base corner, width, depth, height**
- **Base corner (C)**: (-*wa*/2, -*wb*/2, -*wl*)
- **Width (W)**: *wa*
- **Depth (D)**: *wb*
- **Height (H)**: *wl*

5. Set the region of the cuboid to *air*.

6. Select the four faces that represent the waveguide boundary walls and set to PEC.

Select all faces of the waveguide section, except the following faces:

- The face at the origin.
- The face where the FEM modal port will be located (opposite the face at the origin.)

7. Set the solution method for the *air* region to FEM.

> **Tip:** Open the **Modify Region** dialog and click the **Solution** tab. From the **Solution Method** drop-down list, select **Finite Element Method (FEM)**.

8. Create the horn section.

a) Create a flare.

- **Definition method**: **Base centre, width, depth, height, top width, top depth**
- **Bottom width (Wb)**: *wa*
- **Bottom depth (Db)**: *wb*
- **Height (H)**: *hl*
- **Top width (Wt)**: *ha*
- **Top depth (Dt)**: *hb*

9. Delete the face at the origin.

10. Delete the face opposite to the face deleted in Step 9.

11. Union all parts.

12. Add a FEM modal port to the back face of the waveguide.

13. Add a FEM modal source to the port. Use the default settings.

> **Tip:** Exploit model symmetries (if it exists) in a large or complex model to reduce computational costs.

14. Set the frequency to *freq*.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Wire Pin Feed Model.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the same mesh settings as for Wire Pin Feed Model.

   > **Note:** Removing the line (wire) from the model eliminates the need for the **Wire segment radius** specification.

2. Set a local mesh size of `lambda/20` on the back face of the waveguide.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.11.4 Viewing the Results

View and post-process the results in POSTFEKO.

1. Compare the gain (in dB) of the requested E plane radiation pattern on a polar graph.

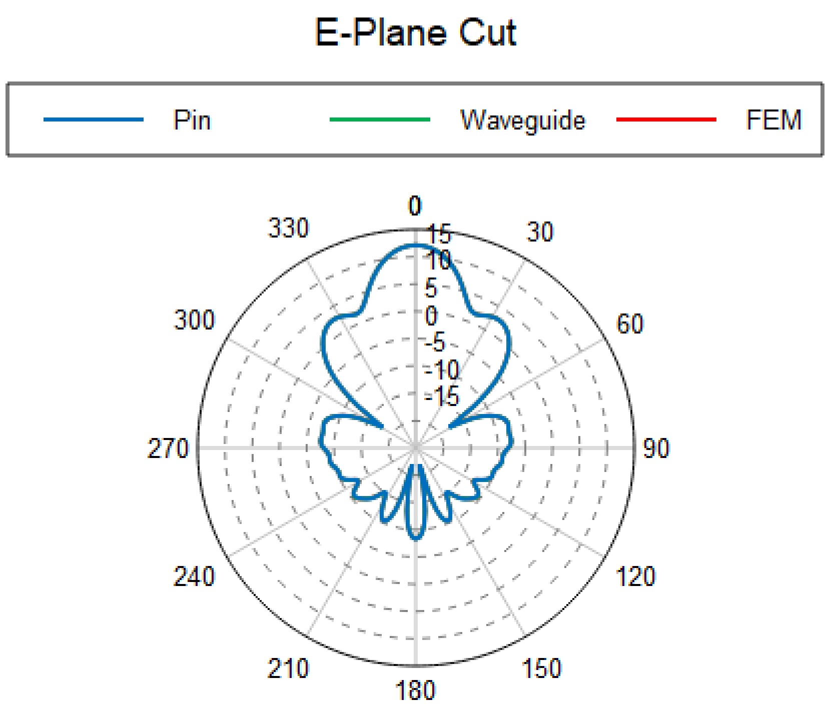


Figure 33: A polar plot of the requested E plane radiation pattern for the horn antenna in POSTFEKO.

2. Compare the gain (in dB) of the requested H plane radiation pattern on a polar graph.

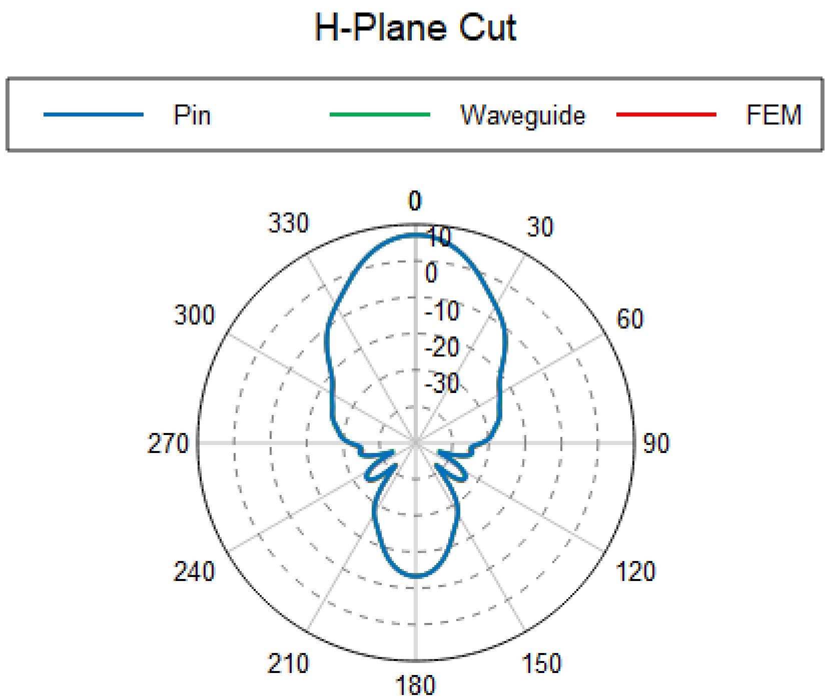


Figure 34: A polar plot of the requested H plane radiation pattern for the horn antenna in POSTFEKO.

A.12 Dielectric Resonator Antenna on Finite Ground

Calculate the input impedance and radiation pattern of a dielectric resonator antenna (DRA) with a coaxial pin feed on a finite ground.

Model the dielectric layers using the following configurations:

1. Feed with a FEM modal source and solve using the hybrid finite element (FEM) and MoM solution.
2. Feed with a waveguide source and solve using the method of moments (MoM) solution with the surface equivalence principle (SEP).

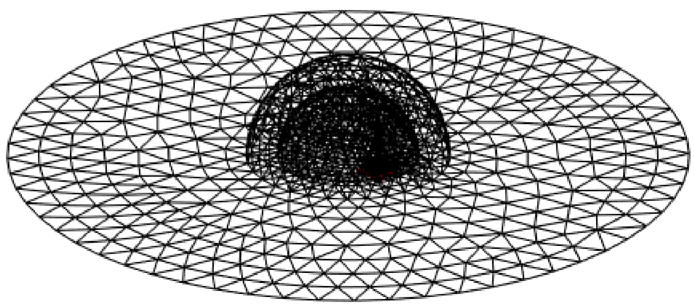


Figure 35: 3D view of the dielectric resonator antenna on a finite ground plane.

A.12.1 Hybrid FEM/MoM Model

Feed the DRA antenna with a FEM modal port. The DRA antenna is solved using the hybrid FEM/MoM method. A layer of air dielectric is added to minimise the number of triangles on the FEM/MoM boundary.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to millimeters.
2. Define the following variables:
   - *epsr* = 9.5 (The relative permittivity of the substrate.)
   - *r* = 0.63 (The radius of the feed element.)
   - *hBig* = 1 (The height of the feed base.)
   - *rBig* = 2.25 (The radius of the feed base.)
   - *rDisk* = 60 (The radius of the ground.)
   - *rDome* = 12.5 (The radius of the inner dome.)
   - *rDomeBig* = *rDome* + 5.5 (The radius of the outer dome.)
   - *h* = 7 (The height of the feed element.)
   - *fmin* = 3e9 (The minimum frequency.)
   - *fmax* = 6e9 (The maximum frequency.)
   - *lambda* = c0/*fmax* * 1000 (The wavelength in free space. [mm])
3. Define a named point:
   - *excite_b*: (0, 6.5, -1)
4. Define a dielectric medium, *air*.
   - **Relative permittivity**: `1`
   - **Dielectric loss tangent**: `0`
   - **Label**: `air`
5. Define a dielectric medium, *dome*.
   - **Relative permittivity**: *epsr*
   - **Dielectric loss tangent**: `0`
   - **Label**: `dome`
6. Define a dielectric medium, *isolator*.
   - **Relative permittivity**: `2.33`
   - **Dielectric loss tangent**: `0`
   - **Label**: `isolator`
7. Create a new workplane.
   - **Origin**: `excite_b`
   - Set the new workplane as the default workplane.

8. Create the outer conductor for the feed pin.
   a) Create a cylinder.
      - **Radius (R)**: *rBig*
      - **Height (H)**: *hBig*
      - **Label**: *FeedBase*
9. Create the inner conductor for the feed pin.
   a) Create a cylinder.
      - **Radius (R)**: *r*
      - **Height (H)**: *hBig* + h
      - **Label**: *FeedPin*
10. Set the region that makes up the feed base to *isolator*.
11. Set the default workplane to **Global XY**.
12. Create the finite ground plane.
    a) Create an ellipse.
       - **Radius (U)**: *rDisk*.
       - **Radius (V)**: *rDisk*.
13. Create the inner dome.
    a) Create a sphere.
       - **Radius**: *rDome*.
       - **Label**: *InnerDome*.
14. Create the outer dome.
    a) Create a sphere.
       - **Radius**: *rDomeBig*.
       - **Label**: *OuterDome*.
15. Union all parts and rename the *Union* to *DRA*.
16. Delete the bottom faces of each dome.
17. Set the region of the inner dome to *dome*.
18. Set the region of the outer dome to *air*.
19. Reset any suspect regions to their correct media.

    > **Tip:** Suspect regions are overlapping regions joined in a union.

20. Set the solution method for all regions to FEM.

    > **Tip:** Open the **Modify Region** dialog and click the **Solution** tab. From the **Solution Method** drop-down list, select **Finite Element Method (FEM)**.

21. Set all faces in the model to PEC.
22. Set the following faces to the **Default** face medium.
    - The top and bottom faces of *FeedBase*.

- The remaining dome faces.

23. Add a FEM modal port to the bottom face of *FeedBase*.
24. Add a FEM modal source to the FEM modal port.
25. Set a continuous frequency range from *fmin* to *fmax*.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a vertical far field request (-180°≤θ≤180°, with ϕ=0°). Sample the far field at θ=2° steps.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Mesh size** equal to **Coarse**
2. Set a *lambda* local mesh size on the face of the outer dome.

> **Tip:** The mesh on the outer dome contributes a large portion to the total memory. Coarsen the mesh to reduce the memory requirement.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.12.2 SEP Model

Feed the DRA antenna with a waveguide port. The DRA antenna is solved using the MoM method.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to millimeters.
2. Define the following variables:
   - *epsr* = 9.5 (The relative permittivity of the substrate.)
   - *r* = 0.63 (The radius of the feed element.)
   - *hBig* = 1 (The height of the feed base.)
   - *rBig* = 2.25 (The radius of the feed base.)
   - *rDisk* = 60 (The radius of the ground.)
   - *rDome* = 12.5 (The radius of the inner dome.)
   - *rDomeBig* = *rDome* + 5.5 (The radius of the outer dome.)
   - *h* = 7 (The height of the feed element.)
   - *fmin* = 3e9 (The minimum frequency.)
   - *fmax* = 6e9 (The maximum frequency.)
   - *lambda* = c0/*fmax* * 1000 (The wavelength in free space. [mm])
3. Define a named point:
   - *excite_b*: (0, 6.5, -1)
4. Define a dielectric medium, *dome*.
   - **Relative permittivity**: *epsr*
   - **Dielectric loss tangent**: `0`
   - **Label**: `dome`
5. Define a dielectric medium, *isolator*.
   - **Relative permittivity**: `2.33`
   - **Dielectric loss tangent**: `0`
   - **Label**: `isolator`
6. Create a new workplane.
   - **Origin**: `excite_b`
   - Set the new workplane as the default workplane.
7. Create the outer conductor for the feed pin.
   a) Create a cylinder.
      - **Radius (R)**: *rBig*
      - **Height (H)**: *hBig*
      - **Label**: *FeedBase*

8. Create the inner conductor for the feed pin.

a) Create a cylinder.

- **Radius (R)**: *r*
- **Height (H)**: *hBig* + h
- **Label**: *FeedPin*

9. Set the default workplane to **Global XY**.

10. Create the finite ground plane.

a) Create an ellipse.

- **Radius (U)**: *rDisk*.
- **Radius (V)**: *rDisk*.

11. Create the inner dome.

a) Create a sphere.

- **Radius**: *rDome*.
- **Label**: *InnerDome*.

12. Union all parts and rename the *Union* to *DRA*.

13. Delete the bottom face of the dome.

14. Set the region that makes up the feed base to *isolator*.

15. Set the region of the inner dome to *dome*.

16. Set all the faces in the model to PEC.

17. Set the following faces to the **Default** face medium.

- The top and bottom faces of *FeedBase*.
- The remaining face of the dome.

18. Add a waveguide port to the bottom face of *FeedBase*.

19. Add waveguide source to the waveguide port.

20. Set a continuous frequency range from *fmin* to *fmax*.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Define the same calculation requests as for Hybrid FEM/MoM Model.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Mesh size** equal to **Coarse**.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.12.3 Viewing the Results

View and post-process the results in POSTFEKO.

1. Compare the input reflection coefficient (in dB) of both methods on a Cartesian graph.

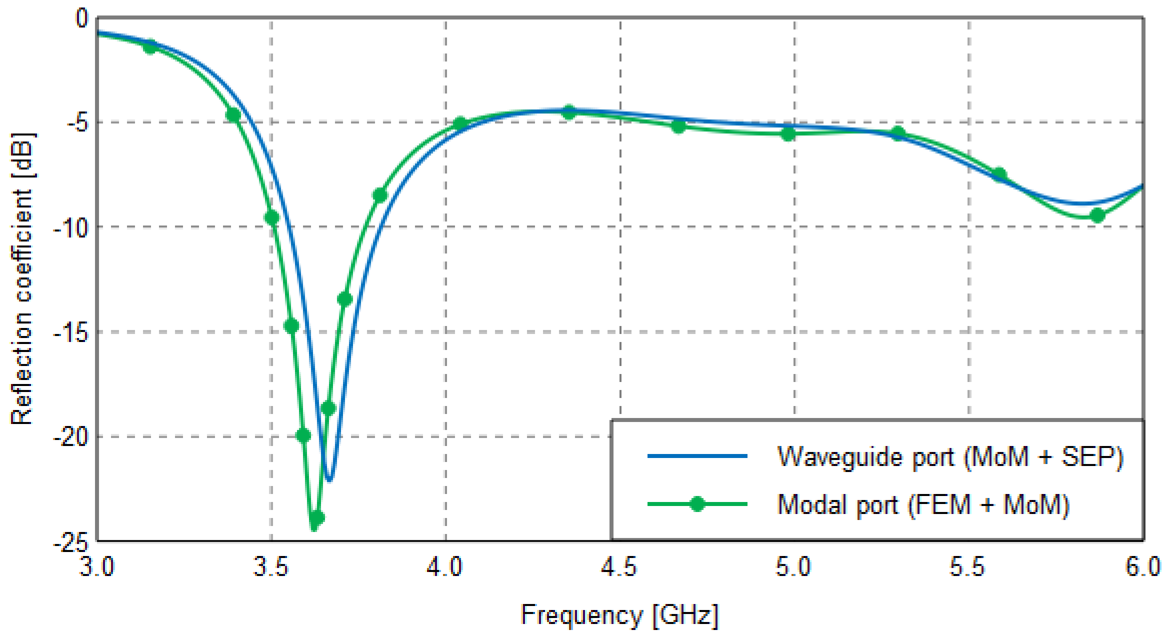


Figure 36: The input reflection coefficient of the DRA for both methods over the operating band.

2. Compare the vertical gain (in dB) of both methods on a polar graph.

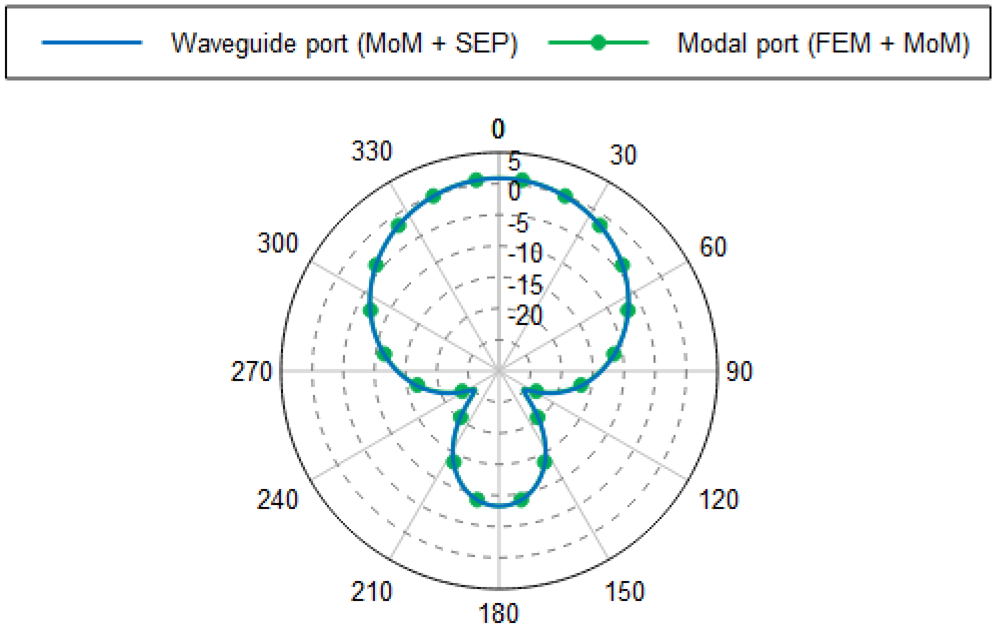


Figure 37: A polar plot of the vertical gain for the DRA at 3.6 GHz for both methods.

A.13 Dielectric Lens Antenna

Calculate the radiation pattern of a dielectric lens antenna. The lens is illuminated by an equivalent far field source with an ideal cosine pattern. The lens structure is modelled using the ray launching geometrical optics (RL-GO). Compare the RL-GO solution with a hybrid FEM/MoM solution.

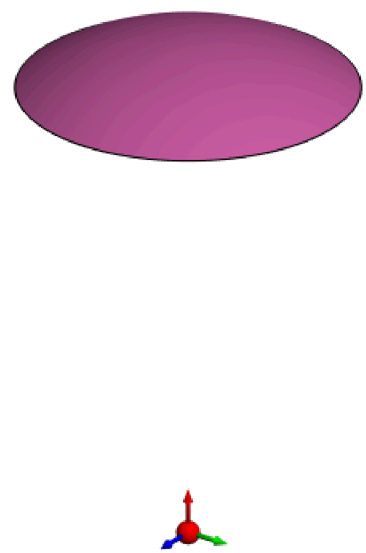


Figure 38: The 3D view of the dielectric lens model with an equivalent far field source.

A.13.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

> **Note:** Assume the focal point of the lens is located at the global origin.

1. Define the following variables.
    - *freq* = 30e9 (The operating frequency.)
    - *epsr* = 6 (relative permittivity.)
    - *tand* = 0.005 (dielectric loss tangent.)
    - *lambda_0* = *c0*/*freq* (The wavelength in free space.)
    - *D* = `lambda_0`*10 (lens diameter.)
    - *F* = 1.5**D* (focal length.)
2. Define the following derived variables for the model construction.
    - *alpha* = arcsin(*D*/(2**F*)) (The included angle to the edge of the lens.)
    - *arclength* = *alpha** *F* (The arc length to the edge of the lens.)

- *n* = sqrt(*epsr*) (The refraction index of the lens.)
- *T* = (2*F - sqrt(4*F^2 - D^2))/(2*(n-1)) (The thickness of the length.)
- *v0* = `(F + T) / (n + 1)` (The ellipse offset distance.)
- *u0* = `sqrt(n^2 - 1) * v0` (The diameter of the lens.)
- *w0* = n*v0 (The major axis length of the ellipse.)

3. Define a dielectric medium, *glass*.
   - **Relative permittivity**: *epsr*
   - **Dielectric Loss tangent**: *tand*
   - **Label**: `Glass`

Construct the lens by subtracting a sphere from an elliptical spheroid.

4. Create a sphere.
   - **Definition method**: **Centre, radius**
   - **Centre**: (0, 0, 0)
   - **Radius**: *F*
5. Create the elliptical spheroid.
   a) Create a sphere.
      - **Definition method**: **Centre, radius U, radius V, radius N**
      - **Centre**: (0, 0, *v0*)
      - **Radius (Ru)**: *u0*
      - **Radius (Rv)**: *u0*
      - **Radius (Rn)**: *w0*
6. Subtract the sphere from the elliptical spheroid.
   a) Rename **Subtract1** to `Lens`.

A closed region is by default set to perfect electric conductor (PEC).

7. Set the region of *Lens* to *Glass*.
8. Set the solver method for the dielectric lens antenna to use RL-GO.

   Tip: Open the **Modify Face** dialog and click the **Solution** tab. From the **Solve with special solution method** list, select **Ray launching - geometrical optics (RL-GO)**.

9. Set the frequency to *freq*.

The dielectric lens is illuminated by a far field pattern source. The E-field pattern is described by the following equation.

$$E_x = \cos^4(\theta) \text{ where } 0 \le \theta \le \frac{\pi}{2} \text{ is the polar angle from the Z axis.} \qquad (1)$$

10. Define the far field data.
    - **Load field data from a Feko Solver (*.ffe) file**
    - **File name**: `Ideal_CosineQ4_Xpol.ffe`
    - Select **Use all data blocks**
    - **Label**: *FarFieldData1*

11. Create a far field equivalent source using the far field definition, *FarFieldData1*.
    - **Magnitude scale factor**: `1`
    - **Phase offset (degrees)**: `0`
    - **Field data**: *FarFieldData1*.

> **Note:** The far field source is positioned at the origin which coincides with the focal point of the lens.

A.13.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Directivity is derived from gain by removing losses. Losses cannot be calculated in an RL-GO dielectric.

1. Create a vertical far field request (-180°≤θ≤180°, with ϕ=0°). Sample the far field at θ=0.25° steps.
   a) Modify the far field request to calculate gain.

   > **Note:** Open the **Request/Modify far fields** dialog, click the **Advanced** tab and then click **Gain**.

2. Create a vertical far field request (-180°≤θ≤180°, with ϕ=90°). Sample the far field at θ=0.25° steps.
   a) Modify the far field request to calculate gain.

   > **Tip:** This setting only applies to the .out file data.

A.13.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Mesh size** equal to **Fine**.

> **Tip:** Curvilinear mesh triangles are created by default to accurately represent the curved geometry. The RL-GO solution method requires an accurate geometric representation only, which is independent of the solution frequency.

A.13.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.

2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.13.5 Viewing the Results

View and post-process the results in POSTFEKO.

1. Compare the gain (in dB) of the requested far field pattern on a polar plot.

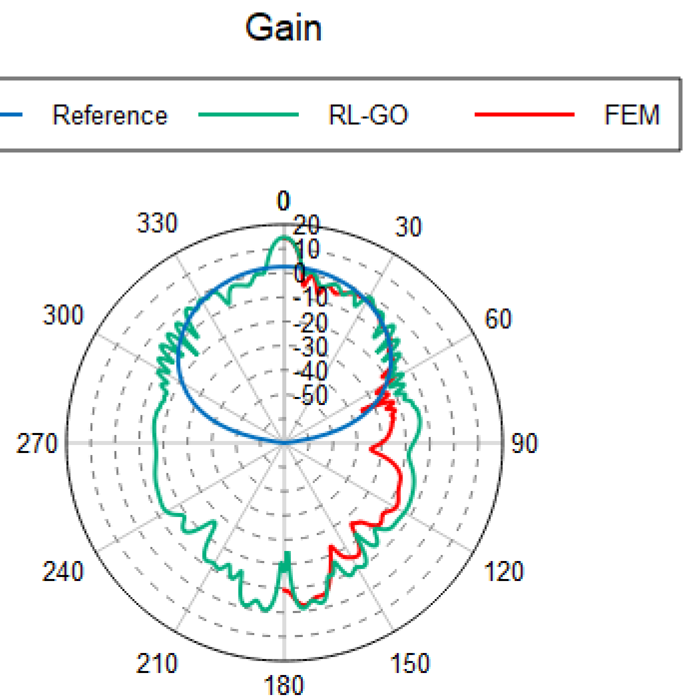


Figure 39: A polar plot of the requested radiation pattern compared to a FEM/MoM solution and the far field source in isolation.

> **Note:** The gain pattern of the equivalent source (labelled *Reference*) is included by importing the .ffe file. Used as a reference, it is compared to the lens antenna solved with RL-GO and the hybrid FEM/MoM.

> **Tip:** The memory and runtime requirements for the RL-GO solution are substantially lower than the FEM/MoM.

2. Optional: Add an image of the lens to the polar graph.

A.14 Windscreen Antenna on an Automobile

Calculate the input impedance of a windscreen antenna constructed with wires. The windscreen consists of a layer of glass and a layer of foil.

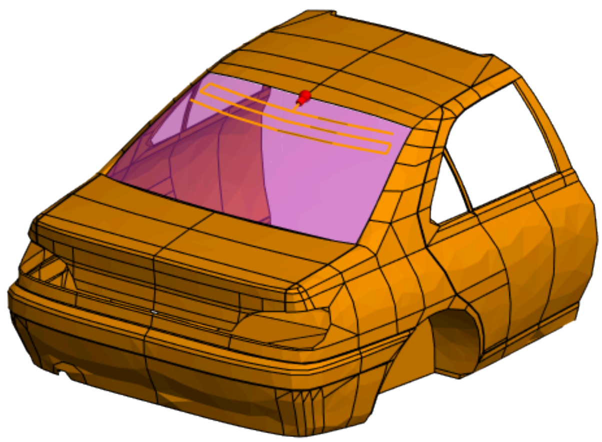


Figure 40: A 3D view of an automobile and windscreen antenna.

The windscreen antenna (wires) can be embedded in the windscreen layers or placed on the surface of the windscreen.

The windscreen curvature reference can consist of multiple layers with different media. Its mesh elements do not contribute to the solution's computational resources.

A.14.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Import the Parasolid geometry of the car from the file `car_geometry.x_b`

   > **Note:** The model is included in the Feko installation.

2. Rename the three imported parts as follows:
   a) The structure for the car: `car_body`.
   b) The structure for the antenna: `antenna`.
   c) The structure for the windscreen: `windscreen`.
3. Union `car_body` and `antenna`.

   > **Note:** The windscreen curvature reference is not part of the union as it is not required to be electrically connected to the model.

4. Add a wire port to the start of the line.
5. Add a voltage source to the port. (1 V, 0°, 50 Ω).
6. Create a dielectric medium (glass).
    - **Relative permittivity**: `7`
    - **Dielectric loss tangent**: `0.03`
    - **Label**: `glass`
7. Create a dielectric medium (PVB foil).
    - **Relative permittivity**: `3`
    - **Dielectric loss tangent**: `0.05`
    - **Label**: `pvb_foil`
8. Create a layered dielectric (2D).
    a) **Label**: `windscreen_layers`
    b) **Layer 1**:
        - **Thickness**: `2.1e-3`
        - **Dielectric material**: `glass`
    c) **Layer 2**:
        - **Thickness**: `0.76e-3`
        - **Dielectric material**: `pvb_foil`
    d) **Layer 3**:
        - **Thickness**: `2.1e-3`
        - **Dielectric material**: `glass`
9. Create a windscreen medium.
    a) **Layer definition**: `windscreen_layers`
    b) **Offset L**: `2.1e-3 + 0.76e-3`
    c) **Label**: `Windscreen1`

    > **Tip:** Variable **Offset L** specifies the reference plane where the windscreen antenna (wires) are located. For this example, the antenna is placed between the two layers, *glass* and *pvb_foil*.

10. Specify the windscreen curvature reference.
    a) Select the single face of the windscreen.
    b) Include the windscreen curvature reference as part of the windscreen solution.

    > **Note:** The windscreen curvature reference is used as part of the windscreen solution to define the shape and position of the windscreen.

    > **Tip:** Open the **Modify Face** dialog, click the **Solution** tab. From the **Solution method** drop-down list, select **Windscreen**.

A windscreen curvature reference is displayed semi-transparent in the colour of the windscreen definition.

11. Specify the windscreen antenna (wires).
   a) Select the windscreen antenna wires.
   b) Include the windscreen antenna as part of the windscreen solution.

> **Note:** Variable **Offset A** is the distance from the windscreen curvature reference to the windscreen antenna (wires). For this example, set **Offset A** equal to `0` to place the antenna on the reference plane.

> **Tip:** Open the **Modify Edge** dialog, click the **Solution** tab. From the **Solution method** drop-down list, select **Windscreen**.

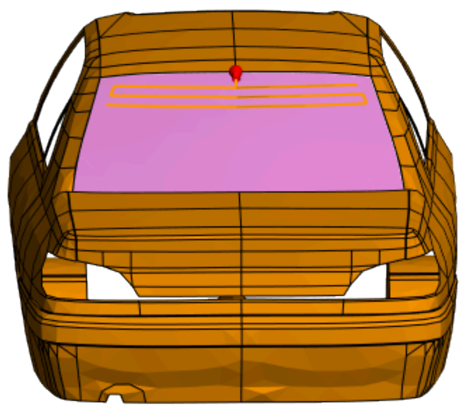


Figure 41: 3D view showing the selected windscreen antenna.

12. Set the continuous frequency range from 90 MHz to 110 MHz.

A.14.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

No solution requests are required.

> **Note:** Input impedance results are always available for voltage sources.

A.14.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Wire segment radius** equal to `150e-6`.
2. Set the local mesh size on the windscreen reference face equal to `0.2`.

> **Note:** Apply a local mesh refinement on the windscreen reference face to ensure accurate representation of the surface. The mesh elements of this face do not contribute to the solution's computational resources.

Due to the fine geometric detail of the car, advanced mesh settings are applied.

3. Specify the advanced mesh settings.
   a) Set the **Refinement factor** equal to **Coarse**.
   b) Set the **Minimum element size** to **Medium**.

> **Tip:** Open the **Modify Mesh Settings** dialog and click the **Advanced** tab.

A.14.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.14.5 Viewing the Results

View and post-process the results in POSTFEKO.

View the antenna input impedance over the operating band on a Cartesian graph.

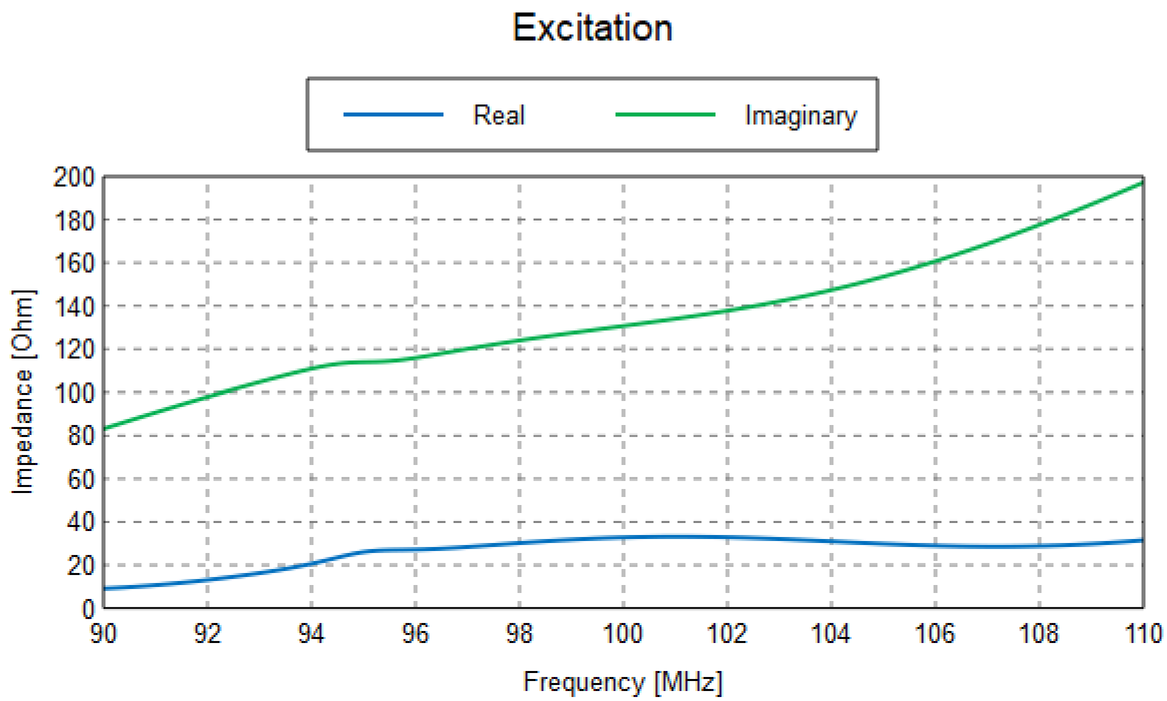


Figure 42: The input impedance (real and imaginary) of the windscreen antenna over the operating band.

Tip: Use MLFMM for solving higher frequencies.

A.15 MIMO Elliptical Ring Antenna (Characteristic Modes)

Calculate the current distribution and far fields for a MIMO elliptical ring antenna. Use characteristic mode analysis to calculate the results for different modes.

The analysis is independent of sources and provides insight into how a structure resonates at the calculated frequencies. This information can be used to excite the structure with the desired modes only.

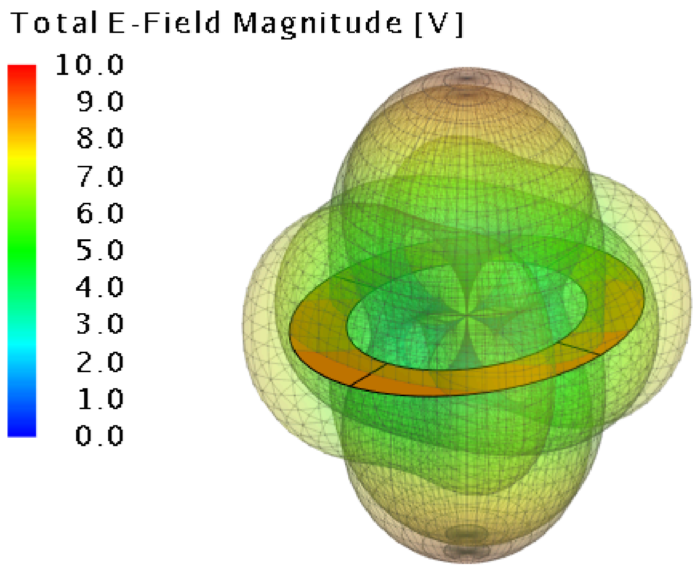


Figure 43: The first four electric far field modes for the MIMO ring.

A.15.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to millimetres.
2. Define the following variables:
   - *rInU* = 21 (The inner radius of the elliptic arc in the U direction.)
   - *rOutU* = 31 (The outer radius of the elliptic arc in the U direction.)
   - *rInV* = 0.8**rInU* (The inner radius of the elliptic arc in the V direction.)
   - *rOutV* = 0.8**rOutU* (The outer radius of the elliptic arc in the V direction.)
   - *freq* = 2.49e9 (The operating frequency.)
3. Create a quarter of the ring.
   a) Create the first elliptic arc.

- **Centre point**: (0, 0, 0)
- **Radius (Ru)**: *rOutU*
- **Radius (Rv)**: *rOutV*
- **Start angle (A0)**: 0°
- **End angle (A1)**: 90°

b) Create the second elliptic arc.

- **Centre point**: (0, 0, 0)
- **Radius (Ru)**: *rInU*
- **Radius (Rv)**: *rInV*
- **Start angle (A0)**: 0°
- **End angle (A1)**: 90°

Create a quarter of the ring antenna.

4. Create a surface from the two elliptic arcs using the **Loft** tool.

a) Rename the label to `sector_1`.

5. Create the full ring antenna.

a) Copy and mirror *sector_1* around the UN plane.

b) Copy and mirror *sector_1* and the copied part from Step 5.a around the VN plane.

c) Union the four sectors to create a single ring structure.

6. Create four edge ports.

a) *port_North* with its port edge on the positive Y axis.

b) *port_East* with its port edge on the positive X axis.

c) *port_South* with its port edge on the negative Y axis.

d) *port_West* with its port edge on the negative X axis.

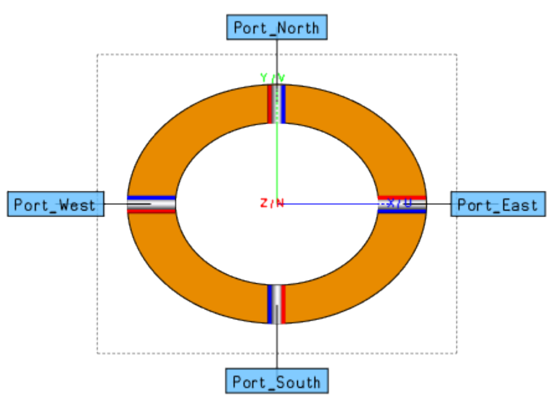


Figure 44: The four edge ports for the ring antenna.

> **Note:** All four ports point in an anticlockwise direction.

7. Set the frequency to *freq*.

8. Specify the symmetry about 2 principal planes.

- X=0: **Geometric symmetry**.
- Y=0: **Geometric symmetry**.

> **Tip:** Exploit model symmetries (if it exists) in a large or complex model to reduce computational costs.

> **Note:** Electric or magnetic symmetry does not apply to characteristic mode analysis, since there are no active sources involved. The geometrical symmetry enforces a symmetrical mesh.

A.15.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create three configurations. The first configuration requests characteristic mode analysis. The second and third configurations excite specific modes.

1. Request a **Characteristic modes** configuration.
   a) **Number of modes to calculate**: 5
   b) Create a currents request (all currents).
   c) Create a full 3D far field request.
2. Request a **Standard configuration**.
   a) Add a voltage source to *port_East*. (1 V, 0°, 50 Ω).
   b) Add a voltage source to *port_West*. (1 V, 180°, 50 Ω).
   c) Create a full 3D far field request.
3. Request a second **Standard configuration**.
   a) Add a voltage source to *port_East*. (1 V, 0°, 50 Ω).
   b) Add a voltage source to *port_West*. (1 V, 0°, 50 Ω).
   c) Add a voltage source to *port_North*. (1 V, 180°, 50 Ω).
   d) Add a voltage source to *port_South*. (1 V, 180°, 50 Ω).
   e) Create a currents request (all currents).
   f) Create a full 3D far field request.

Ensure that sources are specified correctly for the specified standard configurations.

A.15.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Mesh size** equal to **Fine**.

Specify an advanced mesh setting.

2. Set the **Refinement factor** equal to **Fine**.

> **Tip:** The refinement factor ensures the geometry is accurately represented for the higher order modes.

A.15.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.15.5 Viewing the Results

View and post-process the results in POSTFEKO.

When you add excitations or loads to a solution, you unknowingly calculate the weighted sum of the various characteristic modes. Characteristic modes allow you to alter the behaviour of a structure without making any changes to the geometry.

1. Observe how the first characteristic mode can be recreated when sources are placed in the appropriate locations.

   > **Note:** When comparing the characteristic modes to the reconstructed modes, all values need to be normalised.

   a) Plot the currents for the **CharacteristicModesConfiguration1** (mode index = 1) in the 3D view.
   b) Plot the currents for **StandardConfiguration1** in a second 3D view.
   c) Compare the currents for the first mode in **CharacteristicModesConfiguration1** with the reconstructed mode using sources placed in the appropriate locations (**StandardConfiguration1**).

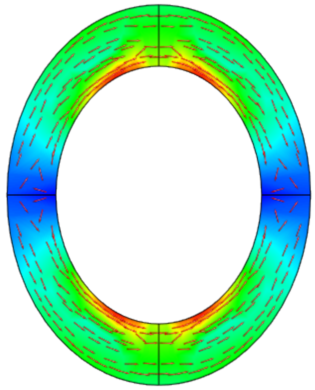


Figure 45: The first characteristic mode of the MIMO ring.

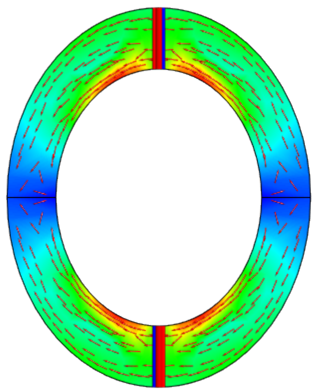


Figure 46: The first reconstructed mode of the MIMO ring.

2. Observe how the fifth characteristic mode can be recreated when sources are placed in the appropriate locations.

a) Plot the currents for the **CharacteristicModesConfiguration1** (mode index = 5) in a third 3D view.

b) Plot the currents for **StandardConfiguration2** in a fourth 3D view.

c) Compare the currents for the fifth mode in **CharacteristicModesConfiguration1** with the reconstructed mode using sources placed in the appropriate locations (**StandardConfiguration2**).

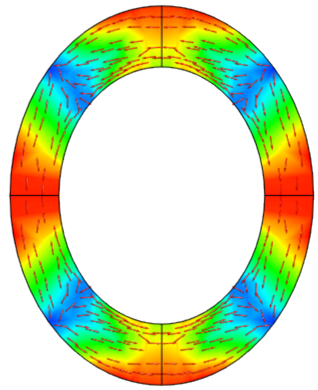


Figure 47: The fifth characteristic mode of the MIMO ring.

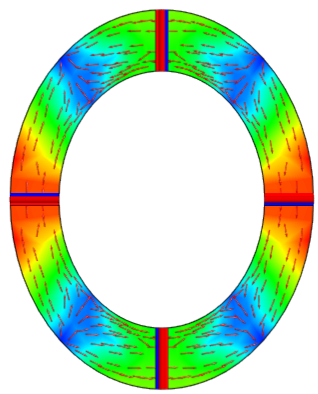


Figure 48: The fifth reconstructed mode of the MIMO ring.

3. Compare the far fields of the characteristic modes configuration and the reconstructed modes on a polar graph.

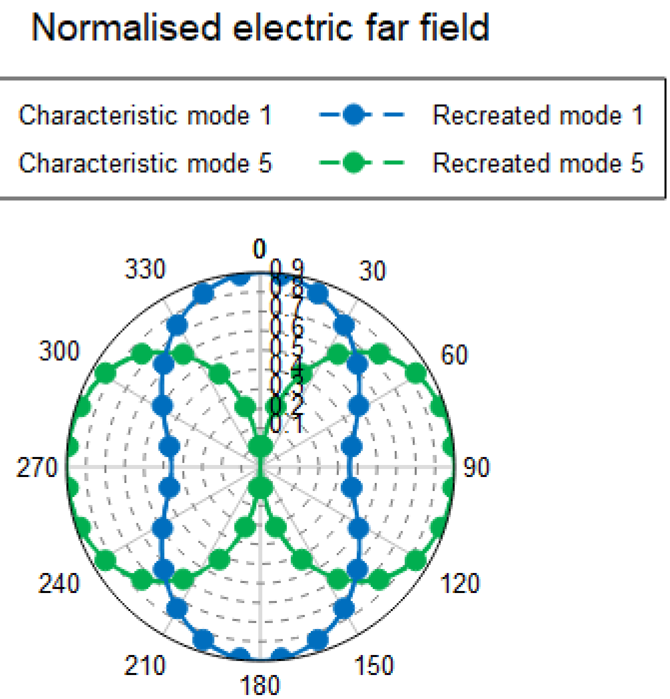


Figure 49: Comparison of the electric fields for the characteristic modes with the reconstructed modes.

> **Note:** Results are normalised in the comparison due to a lack of sources.

The manually excited electric fields are in excellent agreement with the electric fields from the characteristic modes.

A.16 Periodic Boundary Conditions for Array Analysis

Calculate the far field pattern for a single element in an infinite two-dimensional array of pin-fed patch elements. The infinite patch array is modelled using periodic boundary condition. Calculate the approximated far field pattern for a 10x10 element array.

The mutual coupling between elements are taken into account when using periodic boundary condition to model an infinite array. If edge effects can be neglected, use the periodic boundary condition to model a large array accurately.

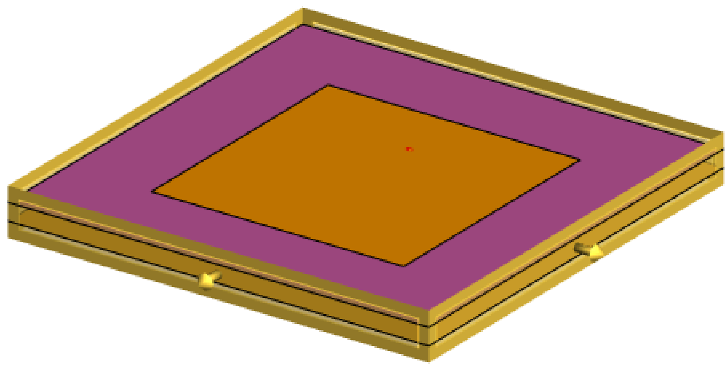


Figure 50: A 3D view of a single element in an infinite patch array in CADFEKO.

> **Tip:** Each model uses its predecessor as a starting point. Create the models in their presentation order. Save each model to a new location to keep them.

A.16.1 Pin-Fed Patch: Broadside Pattern by Phase Shift Definition

Compute the broadside pattern for a single element in an infinite patch array and for a 10x10 element array. The phase shift is specified in the u1 and u2 vector directions.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
   - *lambda* = 0.1 (The spacing for periodic boundary conditions.)
   - *freq* = *c0*/*lambda* (The operating frequency.)
   - *epsr* = 2.55 (Relative permittivity of the substrate.)
   - *base_width* = 0.5**lambda* (Width of the substrate.)
   - *base_length* = 0.5**lambda* (Length of the substrate.)
   - *base_height* = 0.02**lambda* (Height of the substrate.)
   - *patch_width* = 0.3**lambda* (Width of the patch antenna.)
   - *patch_length* = 0.3**lambda* (Length of the patch antenna.)
   - *pin_pos* = *patch_length*/4 (Distance of feed pin from patch centre.)
2. Create a new dielectric called `substrate` with relative permittivity set to *epsr* and the dielectric loss tangent to `0`.
3. Create the substrate.
   a) Create a cuboid.
      - **Definition method**: **Base centre, width, depth, height**
      - **Base corner**: (0, 0, 0)
      - **Width**: *base_width*
      - **Depth**: *base_length*
      - **Height**: *base_height*
4. Create the patch.
   a) Create a rectangle.
      - **Definition method**: **Base centre, width, depth, height**
      - **Base centre**: (0,0, *base_height*)
      - **Width**: *patch_length*
      - **Depth**: *patch_width*
5. Create the feed pin.
   a) Create a wire between the patch and the bottom of the substrate.
      - **Start point**: (-*pin_pos*, 0, 0)
      - **End point**: (-*pin_pos*, 0, *base_height*)
6. Union all elements and label the union `antenna`.

7. Set the region of the cuboid to *substrate*.
8. Set the face of the patch and the bottom face of the substrate to PEC.
9. Add a wire port (segment) to the middle of the line.
10. Add a voltage source to the port. (1 V, 0°, 50 Ω).
11. Set the frequency to *freq*.
12. Set the periodic boundary conditions of the model to end exactly on the edge of the substrate to expand in both the X direction and Y direction.
13. Specify the phase shift for the two directions to **u1**=`0`° and **u2**=`0`°.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Create a vertical far field request (-180°≤θ≤180°, with ϕ=0°). Sample the far field at θ=1° steps.
2. Request the far field calculation for an array of 10x10 elements.
   a) Create a vertical far field request (-180°≤θ≤180°, with ϕ=0°). Sample the far field at θ=1° steps.

   > **Tip:** Open the **Request/Modify far fields** dialog, click the **Advanced** tab and then click the **Calculate far field for an array of elements** check box.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to `0.0001`.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.16.2 Pin-Fed Patch: Broadside Pattern by Squint Angle Definition

Compute the broadside pattern for a single element in an infinite patch array and for a 10x10 element array. The phase shift is determined from the direction into which the beam is pointing ("squint angle").

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Pin-Fed Patch: Broadside Pattern by Phase Shift Definition and rename the file.
2. Modify the periodic boundary conditions.
    a) Determine the phase shift by setting the beam angle for **Theta** and **Phi** to `0°`.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Pin-Fed Patch: Broadside Pattern by Phase Shift Definition.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for Pin-Fed Patch: Broadside Pattern by Phase Shift Definition.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.16.3 Pin-Fed Patch: Squint Pattern by Phase Shift Definition

Compute the squint pattern for a single element in an infinite patch array and for a 10x10 element array. The phase shift is specified in the u1 and u2 vector directions.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Pin-Fed Patch: Broadside Pattern by Phase Shift Definition and rename the file.
2. Modify the periodic boundary conditions.
    a) Set the phase shift for the two directions to **u1**=`-61.56`° and **u2**=`0`°.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Pin-Fed Patch: Broadside Pattern by Phase Shift Definition.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for Pin-Fed Patch: Broadside Pattern by Phase Shift Definition.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.16.4 Pin-Fed Patch: Squint Pattern by Squint Angle Definition

Compute the broadside pattern for a single element in an infinite patch array as well for a 10x10 element array. The phase shift is determined from the direction into which the beam is pointing ("squint angle").

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Pin-Fed Patch: Broadside Pattern by Phase Shift Definition and rename the file.
2. Modify the periodic boundary conditions.
   a) Determine the phase shift by setting the beam angle for **Theta**=`20°` and **Phi**=`0°`.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Pin-Fed Patch: Broadside Pattern by Phase Shift Definition.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for Pin-Fed Patch: Broadside Pattern by Phase Shift Definition.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

A.16.5 Viewing the Results

View and post-process the results in POSTFEKO.

Compare the gain (in dB) of the requested far field patterns for a single patch antenna and for the 10x10 element array.

Note:

- The single patch antenna model includes the mutual coupling between the elements as if the patch antenna is in an infinite array.
- The gain is about 20 dB higher for the 10x10 element array than for the single element.

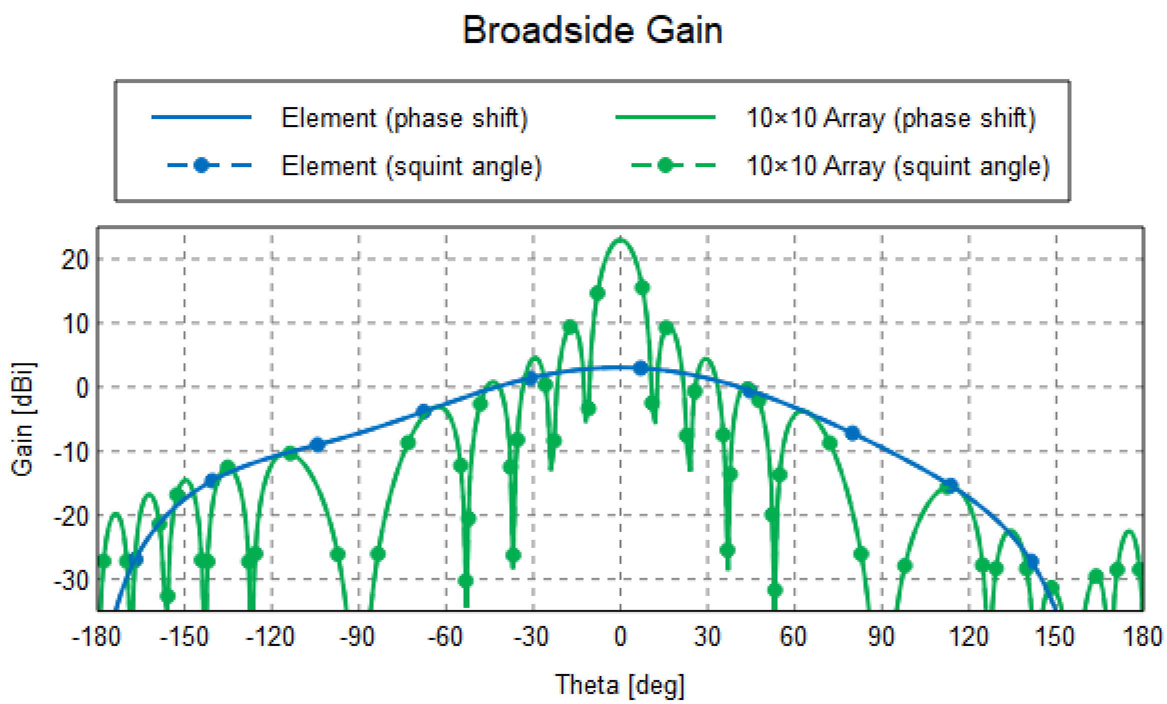


Figure 51: The far field gain for a single element and a 10×10 element patch array in the broadside direction.

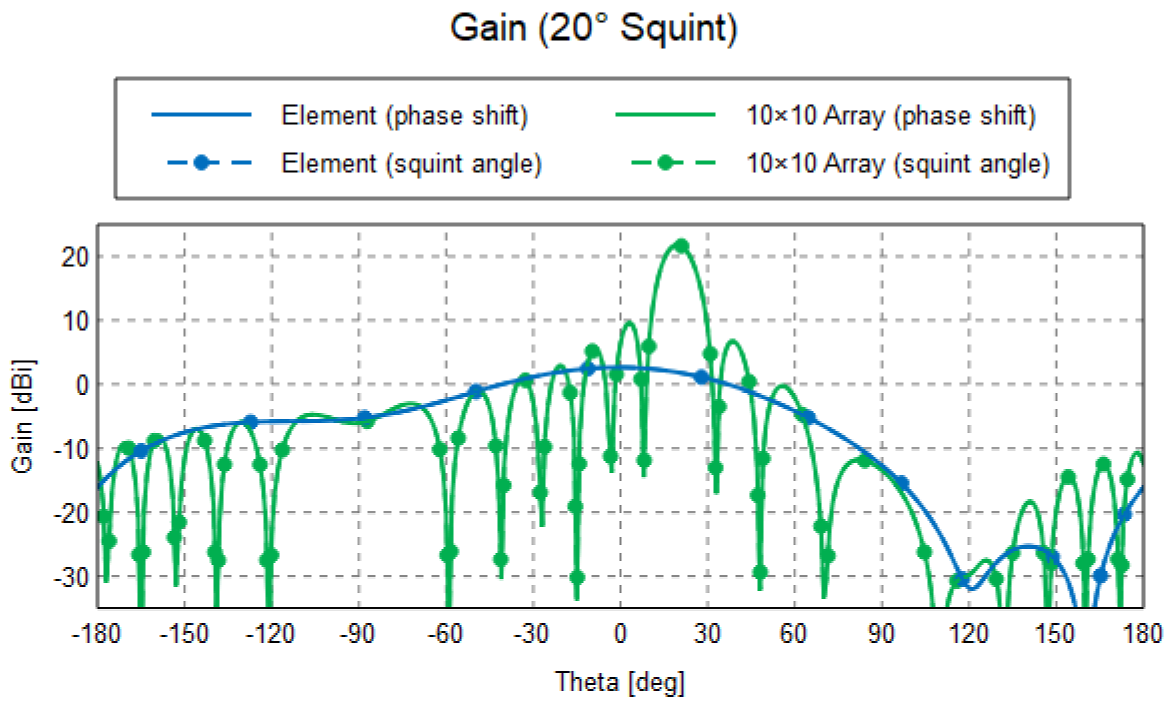


Figure 52: The far field gain for a single element and a 10×10 element patch array in the 20° squint direction.

A.17 Finite Antenna Array with Non-Linear Spacing

Calculate the radiation pattern for an array of arbitrarily placed pin-fed patch antennas. Use the finite array tool to construct the array and the domain Green's function method (DGFM) to minimize computational resources.

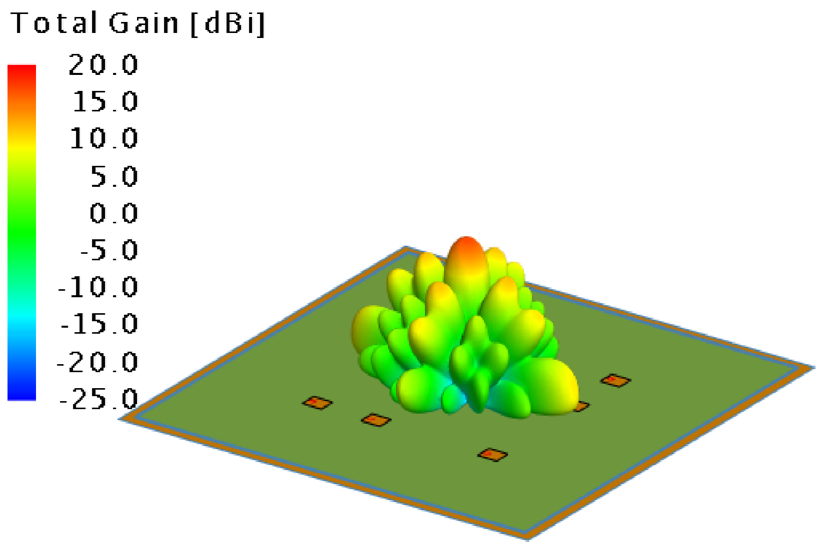


Figure 53: A 3D view of the finite antenna array with far field pattern in POSTFEKO

A.17.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to millimetres.
2. Define the following variables:
   - *freq* = 2.4e9 (The operating frequency.)
   - *lam0* = *c0*/*freq**1000 (The wavelength in free space.)
   - *epsr* = 2.08 (Relative permittivity of the substrate.)
   - *patchLength* = 41 (The length of the patch antenna.)
   - *patchWidth* = 35 (The width of the patch antenna.)
   - *h* = 3.5 (The height of the substrate.)
   - *pinOffset* = -11 (Distance between the feed pin and patch centre.)
   - *wireRadius* = 0.1 (The radius of the feed pin wire.)
3. Create the patch. The patch is the base element to be used in the finite antenna array.
   a) Create a rectangle.

- **Definition method**: **Base centre, width, depth**
- **Base centre**: (0, 0, 0)
- **Width (W)**: *patchWidth*
- **Depth (D)**: *patchLength*

4. Create the feed line.

a) Create a line.

- **Start point**: (0, *pinOffset*, -*h*)
- **End point**: (0, *pinOffset*, 0)

5. Union all parts in the tree.

6. Create a dielectric medium.

a) **Dielectric loss tangent**: `0`

b) **Relative permittivity**: *epsr*

c) **Label**: `substrate`

7. Add a planar multilayer substrate (infinite plane) with a conducting layer at the bottom.

a) Select **Plane / ground.**

- Select **Planar multilayer substrate** from the **Definition method** drop-down list.
- **Thickness** (Layer 1): *h*
- **Medium** (Layer 1): *substrate*
- **Ground plane** (Layer 1): *PEC*
- **Z value at the top of layer 1**: 0

8. Add a wire port (segment) to the middle of the line.

9. Add a voltage source to the port. (1 V, 0°, 50 Ω).

10. Set the frequency to *freq*.

> **Note:** The steps up to this point represents the base element. The next steps will create an array from the base element.

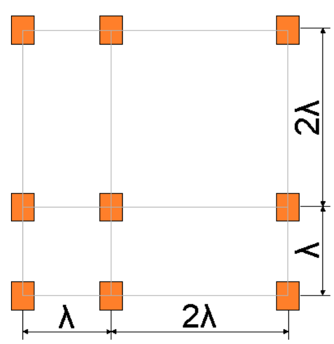


Figure 54: Layout of the final array.

11. Create a planar array.

- **Number of elements** = 4 (in both the U and V dimensions)
- **Offset along X axis** = *lam0*.

- **Offset along Y axis** = *lam0*.

Convert an array into a custom array to allow an element to be rotated or repositioned with respect to one another. Rotate an element by modifying its local workplane. An element can also be deleted from the array.

12. Convert the array into a custom array.

> **Tip:** For this example, the elements are not rotated, but you are encouraged to rotate a few of the elements after obtaining the initial results to investigate the effect on the array pattern.

13. Delete the elements from the third row and third column. Only nine elements should remain.
14. Solve the antenna array with the DGFM.

> **Tip:** Open the **Solver Settings** dialog, click the **Domain Decomposition** tab and then select the **Solve model with Domain Green's Function Method (DGFM)** check box.

A.17.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a full 3D far field request. Sample the far field at θ=1.5° and ϕ=1.5° steps.

a) Change the workplane origin to (1.5**lam0*, 1.5**lam0*, 0) to place the far field at the middle of the antenna array.

A.17.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to *wireRadius*.

A.17.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

Note: The following warning may be encountered when running the Solver:

```
Directivity cannot be computed for far field calculations involving the
planar multilayer Green's function with losses in the dielectric layers,
gain will be computed instead.
```

Losses cannot be calculated in an infinitely large medium as is required for the extraction of antenna directivity information.

Avoid this warning by requesting the far field gain instead of the directivity. Open the **Request/Modify far fields** dialog, click the **Advanced** tab and then click **Gain**.

A.17.5 Viewing the Results

View and post-process the results in POSTFEKO.

1. View the gain (in dB) of the requested far field pattern using a polar plot.
2. Compare the far field pattern of the finite antenna array with the equivalent full MoM model.

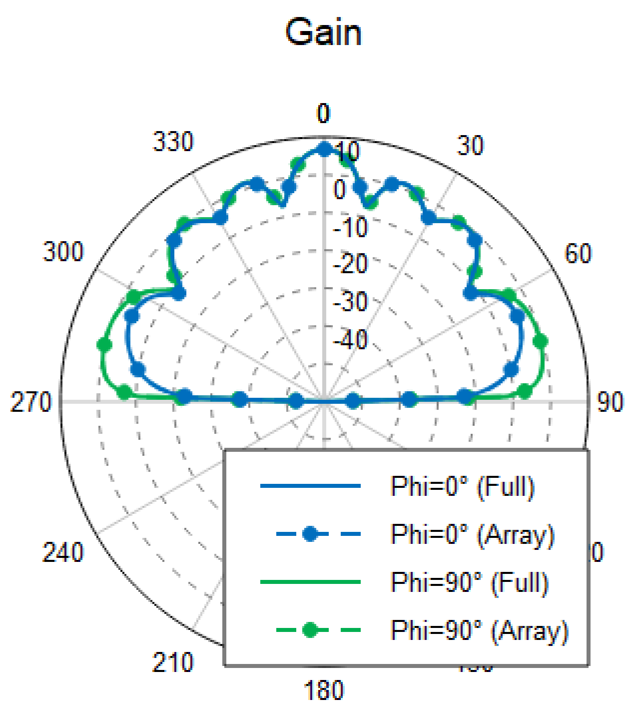


Figure 55: A polar plot of the far field gain (dB) viewed in POSTFEKO. The gain of the finite antenna array is compared to the equivalent full MoM model.

Note: The finite array tool simplifies array construction. For larger arrays, the performance improvement of the DGFM are more pronounced.

Antenna Placement

B

Simple examples demonstrating antenna placement.

This chapter covers the following:

B.1 Antenna Coupling on an Electrically Large Object

Calculate the S-parameters (coupling) over a frequency range for three monopole antennas located near the front, middle and rear of a Rooivalk helicopter mock-up.

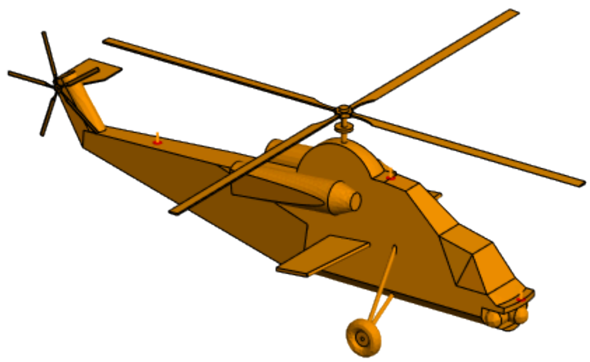


Figure 56: A 3D view of the Rooivalk helicopter with the monopole antennas located near the front, middle and rear.

B.1.1 Creating the Model

> **Note:** This model contains complex geometry. The creation steps are not provided, but the model is included in the Feko installation.

Solve the model with the MLFMM.

> **Tip:** Open the **Solver settings** dialog, click the **MLFMM / ACA** tab and then click **Solve model with the multilevel fast multipole method (MLFMM)**.

B.1.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Request the S-parameters for the model. Add the three ports in the S-parameter request and specify a reference impedance of 50 Ω each. Set the three ports to active.

B.1.3 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

B.1.4 Viewing the Results

View and post-process the results in POSTFEKO.

The resource requirements (time and memory) for the MLFMM solution are notably smaller than for the full MoM solution. Some solution times from the `.out` file are given below. Note that times are strongly dependent on the hardware.

```
                            SUMMARY OF REQUIRED TIMES IN SECONDS

                                                 CPU-time        runtime
Reading and constructing the geometry               1.630          1.631
Calcul. of the MLFMM transfer function              0.876          0.876
Fourier transform of MLFMM basis funct.             4.505          4.508
Calcul. of matrix elements                        126.248        126.251
Calcul. of right-hand side vector                   0.099          0.099
Preconditioning system of linear eqns.             44.603         44.603
Solution of the system of linear eqns.            112.489        112.486
                                             ------------   ------------
                        total times:              293.942        293.941
              (total times in hours:                0.082          0.082)

Specified CPU-times are referring to the master process only
Sum of the CPU-times of all processes:      2351.498 seconds (     0.653 hours)
On average per process:                      293.937 seconds (     0.082 hours)

Peak memory usage during the whole solution: 443.363 MByte
```

1. For the antennas mounted on the helicopter:
   a) View the coupling between the antennas as a function of frequency on a Cartesian graph.

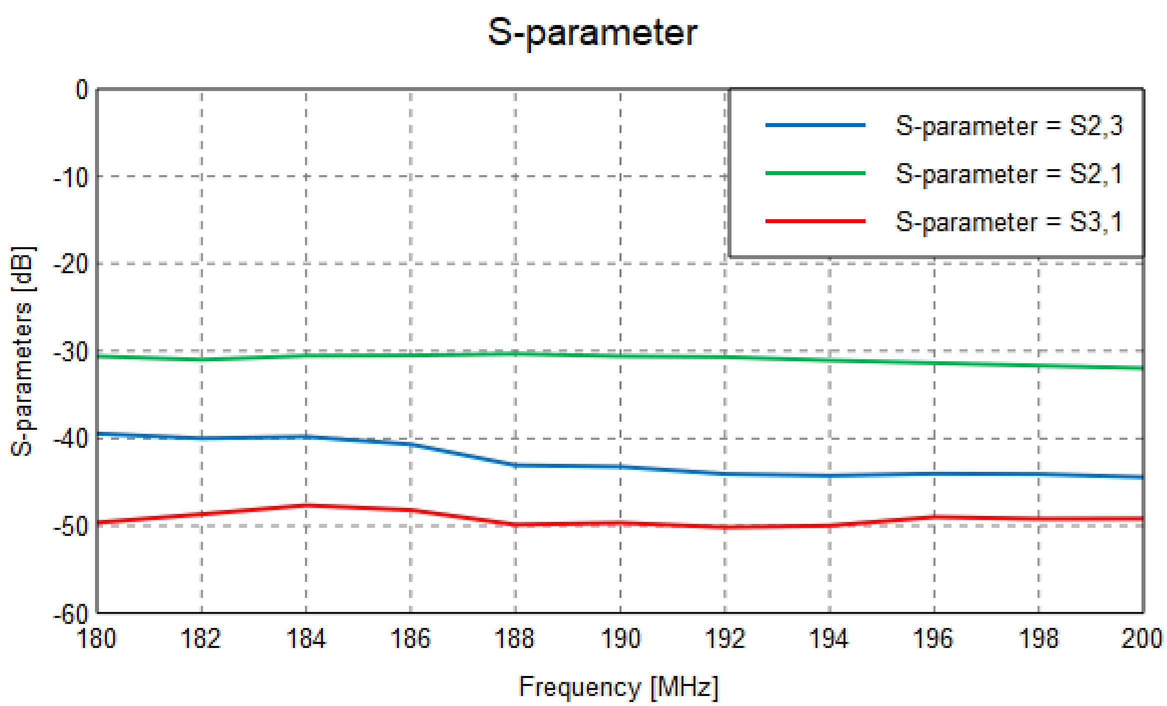


Figure 57: The antenna coupling as a function of frequency on a Cartesian graph.

2. View the reflection coefficients of the antennas as a function of frequency on a Cartesian graph.

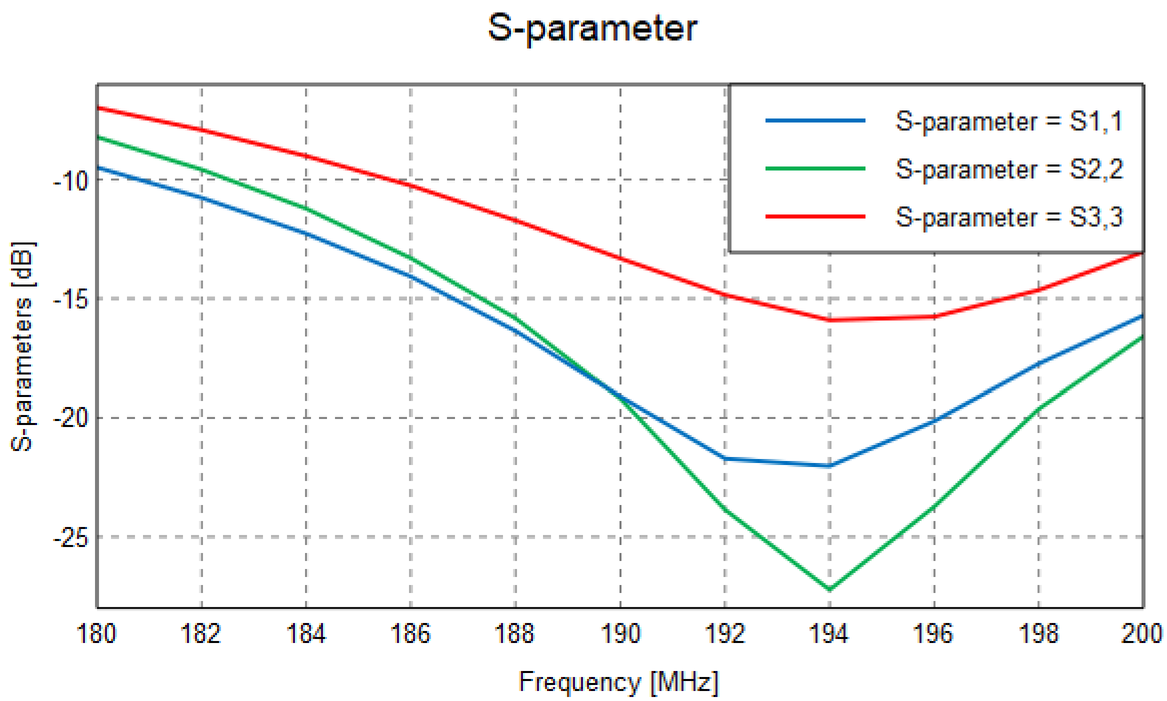


Figure 58: The reflection coefficients of the antennas as a function of frequency on a Cartesian graph.

B.2 Antenna Coupling Using an Ideal Receiving Antenna

Calculate the coupling between a helix antenna and a Yagi-Uda antenna located in front of a large plate. Reduce computational resources by using the uniform theory of diffraction (UTD) and an ideal receiving antenna.

The receiving antenna is modelled with three equivalent field types:

1. far field radiation pattern
2. far field spherical modes
3. near field aperture

The results will be compared for all three field types.

> **Note:** Equivalent field sources and receiving antennas are impressed fields and are not influenced by nearby physical structures.
>
> For accuracy, ensure sufficient distance to the physical structures.

Three models are provided for this example:

- `Antenna_Coupling_Helix_Antenna.cfx`: Model of the helix antenna used to pre-calculate the three field types that will be used in the ideal receiving antennas.
- `Antenna_Coupling_Receiving_Antenna.cfx`: Model that calculates the antenna coupling using the ideal receiving antenna types.
- `Antenna_Coupling_Full.cfx`: Full model for both antennas.

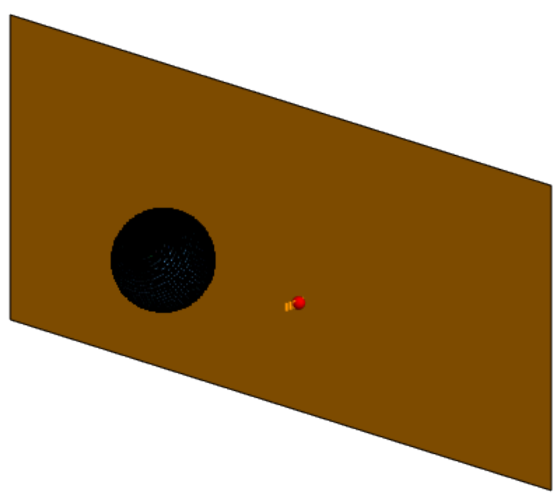


Figure 59: 3D view of the full model.

B.2.1 The Helix Antenna - Full Model

Calculate the near field, far field and spherical modes of a helix antenna. Export the fields to a file to use as an ideal receiving antenna.

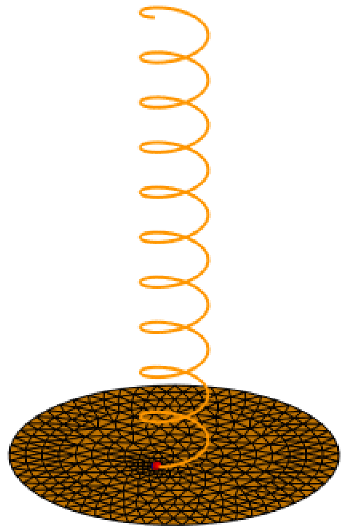


Figure 60: 3D view of the helix antenna.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Create the following variables.
   - *freq* = `1.654e9` (The operating frequency.)
   - *lambda* = `c0/freq` (The wavelength in free space.)
   - *n* = `10` (number of turns for the helix.)
   - *helix_alpha* = `13` (pitch angle of the helix.)
   - *helix_radius* = `lambda*cos(helix_alpha*pi/180)/pi/2` (radius of the helix.)
   - *plate_radius* = `0.75*lambda` (radius of the ground plate.)
   - *wire_radius* = `0.65e-3` (radius of the helix wire segments.)
2. Create the circular base plate of the helix.
   a) Create an ellipse.
      - **Centre point**: `(0, 0, 0)`
      - **Radius (U)**: *plate_radius*
      - **Radius (V)**: *plate_radius*
3. Create the helix

- **Definition method**: **Base centre, radius, pitch angle, turns**
- **Origin**: `(0, 0, 0)`
- **Radius**: *helix_radius*
- **Pitch angle**: *helix_alpha*
- **Number of turns**: *n*

4. Union the helix and ellipse.
5. Add a wire port to the start of the line.
6. Add a voltage source to the port. (1 V, 0°, 50 Ω).
7. Set the frequency to *freq*

Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Create a full 3D far field request.
   a) On the **Advanced** tab of the far field request, enable **Export fields to ASCII file (*.ffe)**.
   b) Enable **Calculate spherical expansion mode coefficients**.
   c) Enable **Export spherical expansion mode coefficients to ASCII file.**

   > **Note:**
   > - Define a far field receiving antenna using a `.ffe` file.
   > - Define a spherical modes receiving antenna using a `.sph` file.

2. Create a near field request.
   a) **Definition method**: `Spherical`
   b) On the **Advanced** tab enable **Export fields to ASCII file (*.efe/*.hfe)**
   c) **Start**: `(0.45, 0, 0)`
   d) **End**: `(0.45, 180, 360)`
   e) **Increment**: `(0, 5, 5)`

   > **Note:** Define a near field aperture receiving antenna using `.efe`/`.hfe` files.
   > Since the near field surface should capture the entire radiating area of the antenna and the helix antenna has a small ground plane, a spherical near field request surrounding the antenna is sufficient.

The exported files are used in the set up of Antenna Coupling with Ideal Receiving Antenna.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to *wire_radius*.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

B.2.2 Antenna Coupling with Ideal Receiving Antenna

Calculate the antenna coupling between a Yagi-Uda and helix antenna. Create the Yagi-Uda antenna and use an ideal receiving antenna in place of the helix antenna.

Use the field data exported to file in The Helix Antenna - Full Model.

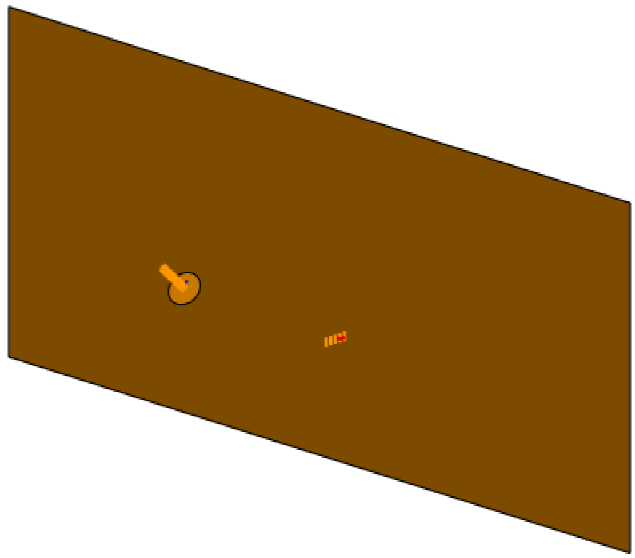


Figure 61: 3D view of the ideal receiving antenna.

Creating the Variables and Named Points

Create the variables and named points used in the construction of the Yagi-Uda antenna.

1. Create the following variables.
   - *freq* = `1.654e9` (The operating frequency.)
   - *lambda* = *c0/freq* (The wavelength in free space.)
   - *yagi_ld* = `lambda * 0.442` (length of the director element.)
   - *yagi_li* = `lambda * 0.451` (length of the active element.)
   - *yagi_lr* = `lambda * 0.477` (length of the reflector element.)
   - *yagi_d* = `0.25 * lambda` (spacing between Yagi elements.)
   - *yagi_rho* = `lambda * 0.0025` (radius of the helix wire segments.)
2. Create the following named points.
   - *helix_centre* = `(-1.5/2, 0.75, 1.5)` (helix antenna location.)
   - *yagi_centre* = `(-1.5/2, -0.75, 1.5)` (Yagi antenna location.)

Constructing the Yagi-Uda Antenna

Build a parametric model of a Yagi-Uda antenna. Use variables and named points already created.

1. Create a line (active element).
    - **Start point**: `(0, 0, -`*`yagi_li`*`/2)`
    - **End point**: `(0, 0, `*`yagi_li`*`/2)`
    - **Label**: `yagi_active`
2. Create a line (director element).
    - **Start point**: `(0, -`*`yagi_d`*`, -`*`yagi_ld`*`/2)`
    - **End point**: `(0, -`*`yagi_d`*`, `*`yagi_ld`*`/2)`
    - **Label**: `yagi_director`
3. Create a line (reflector element)
    - **Start point**: `(0, `*`yagi_d`*`, -`*`yagi_lr`*`/2)`
    - **End point**: `(0, `*`yagi_d`*`, `*`yagi_lr`*`/2)`
    - **Label**: `yagi_reflector`
4. Create another director element.
    - Create a copy of the `yagi_director` element.
    - Translate the copy from `(0, 0, 0)` to `(0, -`*`yagi_d`*`, 0)`.
5. Create another director element.
    - Create a copy of the *first* `yagi_director` element.
    - Translate this new copy from `(0, 0, 0)` to `(0, -2 * `*`yagi_d`*`, 0)`.
6. Union the lines and rename the resulting part to `yagi_antenna`.
7. Place the Yagi antenna in the correct position and orientation.
    a) Rotate the **yagi_antenna** part around the N axis by `-(90+15)°`.
    b) Translate the **yagi_antenna** part from `(0, 0, 0)` to (*yagi_centre*, *yagi_centre*, *yagi_centre*).

Creating a Wire Port and Setting the Frequency

Add a wire port and voltage source to the Yagi-Uda antenna.

1. Add a wire port to the middle of the `yagi_active` element.
2. Add a voltage source to the port. (1 V, 0°, 50 Ω).
3. Set the frequency to *freq*.

Creating the Plate

Create the plate. Apply the UTD solution method to the plate.

1. Create the plate.
    a) Create a rectangle.

- On the **Geometry** tab, from the **Definition methods** drop-down list, select **Base corner, width, depth**
- **Base corner (C)**: `(-3, 0, 0)`
- **Width (W)**: `6`
- **Depth (D)**: `3`
- **Label**: `metal_plate`.
- On the **Workplane** tab, select **Predefined workplane**. From the drop-down list, select **Global YZ**.

2. Set the UTD solution properties on the **metal_plate** part.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Define far field data.
   a) **Number of theta points**: 37
   b) **Number of phi points**: 73
   c) **File name**: Browse for the `.ffe` file

> **Tip:** Click on **Field/Current data** on the **Construct** tab to find all the field data types. Select **Define Far Field Data**.

2. Import the spherical mode data from file.
   a) **File name**: Browse for the `.sph` file.

Define near field data.

3. Select **Define Near Field Data**.
   a) **Source type**: **Load from .efe and .hfe file**.
   b) **E-field file**: Browse for the `.efe` file.
   c) **H-field file**: Browse for the `.hfe` file.
   d) **Coordinate system**: **Spherical**
   e) **Radius (R)**: `0.45`
   f) **Number of points along theta**: `37`
   g) **Number of points along phi**: `73`
   h) On the **Workplane** tab set the following:
      1. Select **Custom workplane**
      2. **Origin**: (`helix_centre`, `helix_centre`, `helix_centre`)
      3. **U vector**: `(1, 0, 1)`.
4. Create a far field receiving antenna request.
   a) **Field data**: **FarFieldData1**
   b) On the **Workplane** tab set the following:
      - Select **Custom workplane**

- **Origin**: (`helix_centre`, `helix_centre`, `helix_centre`)
- **U vector**: `(1, 0, 1)`

5. Create a spherical modes receiving antenna request.
   a) **Field data**: **SphericalModesData1**
   b) Set the **Workplane** identical to that set in the far field receiving antenna request in Step 4.
   c) On the **Advanced** tab select **Use far field approximation**.

   > **Tip:** A spherical modes source requires that no geometry breaches the far field distance of the source. Feko will give a warning if this is the case.

6. Create a near field receiving antenna request.
   a) On the **General** tab select **Combine individual faces**
   b) **Field data**: **NearFieldData1**
7. Set the radiated power to 100W.

   > **Tip:** To set the radiated power, on the **Source/Load** tab select **Power** and select the option **Total source power (no mismatch)**.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to *yagi_rho*.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

B.2.3 Viewing the Results

View and post-process the results in POSTFEKO.

A model of the full solution (no ideal receivers or sources) is provided. The ideal receiving antenna calculates the coupling assuming a matched load. The helix antenna is loaded with its complex conjugate impedance. The power in the load represents the total power received by the antenna. Results can then be compared directly.

The coupling results are derived from the received power data. In POSTFEKO the received power may be plotted on a graph. It is also given in the `.out` file.

Compute the coupling from the following equation:

$$\text{Coupling}_{(dB)} = 10 * log10\left[\frac{\text{Received power}}{100}\right] \tag{2}$$

Table 3: Coupling results for the four models at 1.654 GHz

Model	Received power (mW)	Coupling (dB)	Runtime (s)
Full model	2.66	-45.75	15
Far field pattern	2.56	-45.92	<3
Spherical modes	2.60	-45.82	<3
Near field	2.67	-45.73	<3

For convenience results were computed over a frequency range and plotted on a graph in POSTFEKO.

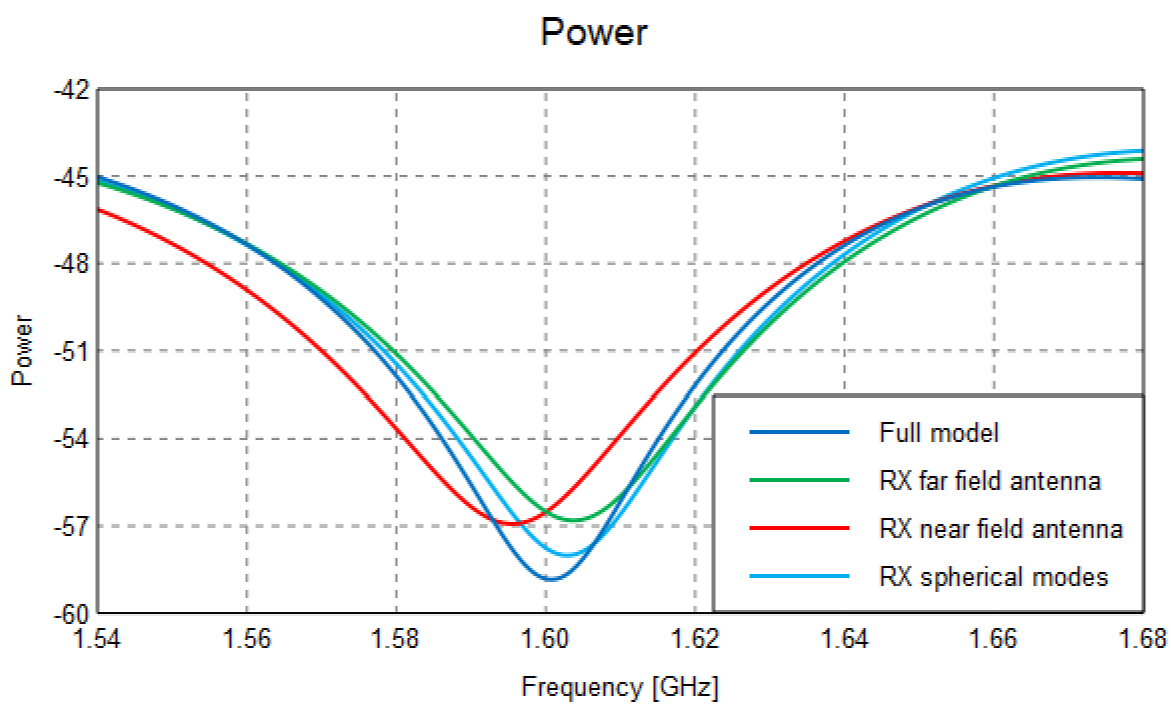


Figure 62: Coupling results over a frequency range for the four models.

Tip: For coupling in dB, use **Enable math** and enter the equation `10*log(self/100)`.

B.3 Antenna Coupling Using an Equivalent Source and Ideal Receiving Antenna

Calculate the coupling between two horn antennas separated by 60 wavelengths. A metallic plate between the horn antennas blocks the line-of-sight coupling. Replace the horn antennas with a far field equivalent source and ideal far field receiving antenna.

Three models are provided for this example. The creation steps are not provided, but the models are included in the installation.

- **Pyramidal_Horn.cfx:** Model of the horn antenna on its own. The far field pattern will be used in the equivalent source and ideal receiving antenna.
- **Full_Model.cfx:** Full model of two horn antennas and plate.
- **Point_Source_Coupling.cfx:** Model that calculates the antenna coupling using the far field equivalent source and ideal receiving antenna.

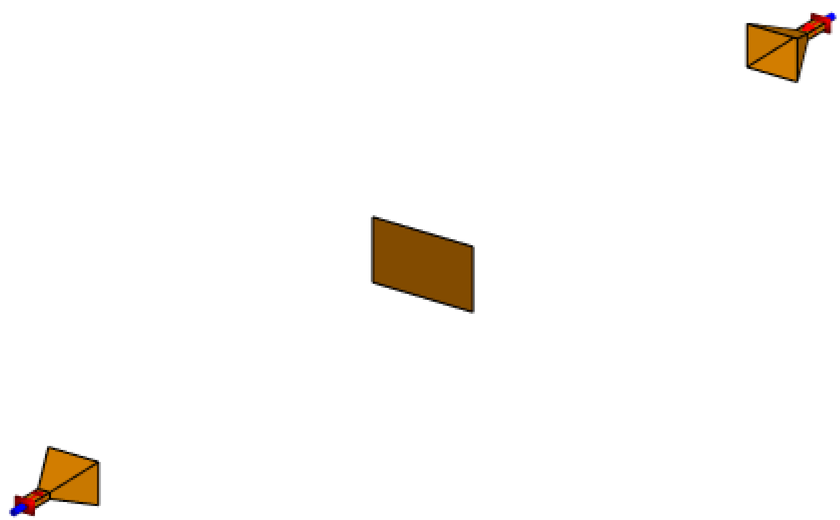


Figure 63: The model using equivalent source and receiving antenna.

B.3.1 Solving the Horn Antenna to Obtain the Fields

Calculate the far field of the horn antenna. Export the fields to file to use as far field equivalent source and ideal receiving antenna.

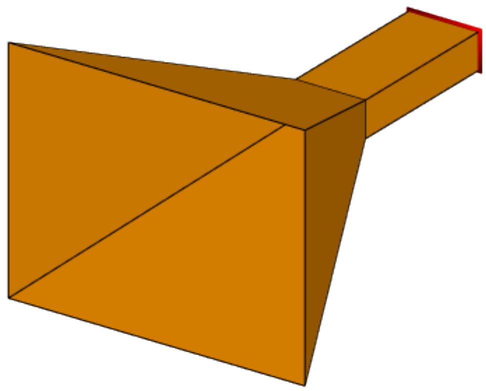


Figure 64: 3D view of the horn antenna.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

The model of the horn is not constructed but provided.

Open `Pyramidal_Horn.cfx`.

Note:

- The waveguide port is placed on the YZ plane and the horn is centered with respect to these axes.
- The phase centre of the horn is located inside the flare. The far field is calculated with the offset axis origin at X = -21.6 cm.
  - The phase centre is required for accurate placement of the equivalent sources, but this calculation is beyond the scope of this example.
  - The phase centre varies over frequency. However, for the narrow bandwidth of this example, this variation is ignored.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

B.3.2 Solving the Equivalent Sources Model

Calculate the antenna coupling between two horn antennas represented by a far field equivalent source and an ideal receiving antenna.

Use the far fields exported to file in the previous section.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

In CADFEKO open `Point_Source_Coupling.cfx`.

> **Note:** The transmitted power is set to 1 W which simplifies the coupling equation to:
>
> $$\text{Coupling}_{(dB)} = 10 * log10\left[\frac{\text{Received power}}{\text{Transmitted power}}\right] \tag{3}$$

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

B.3.3 Obtaining a Full Wave Reference Solution

Calculate the coupling between two horn antennas. Use the full models of both antennas to obtain a reference solution.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

In CADFEKO, open `Full_Model.cfx`.

> **Note:** Full models of both horns are used. The coupling is computed from the S-parameters.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

B.3.4 Viewing the Results

View and post-process the results in POSTFEKO.

1. Create an empty Cartesian graph.
2. Add the **FarFieldReceivingAntenna1** results to the graph.
3. Add the S-matrix results (S21) to graph.

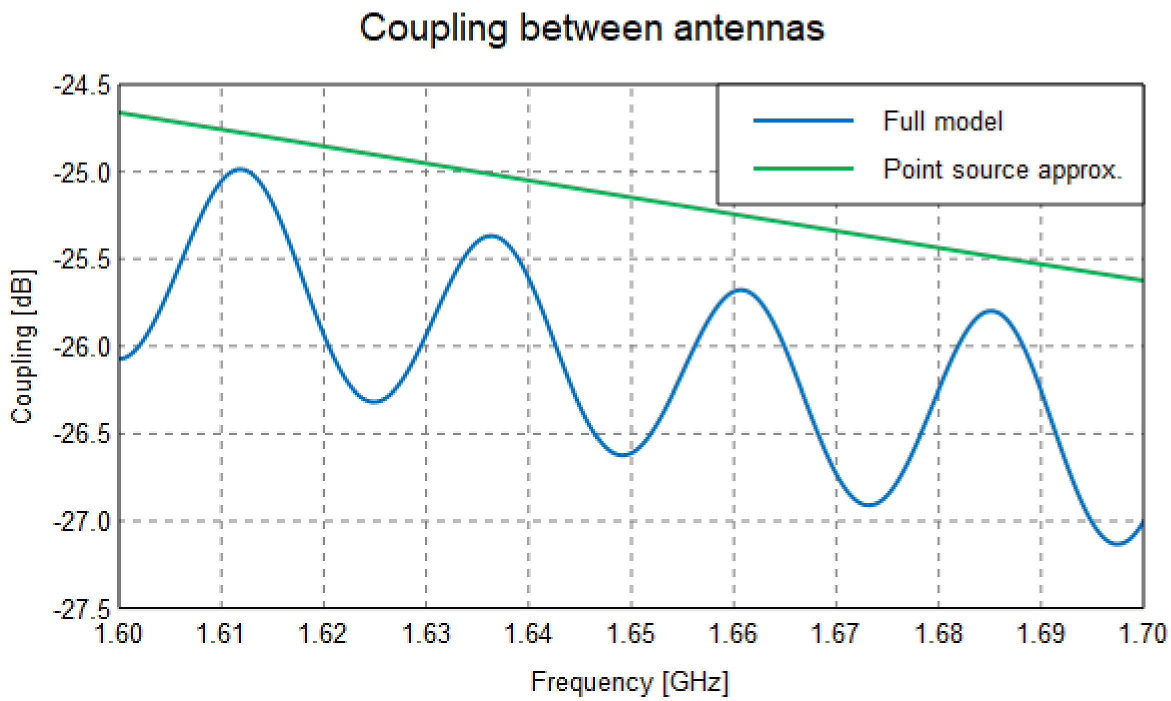


Figure 65: Comparison of the coupling between the antennas using a full solution vs equivalent sources.

Radar Cross Section (RCS)

C

Simple examples demonstrating radar cross section (RCS) calculations of objects.

This chapter covers the following:

C.1 RCS of a Thin Dielectric Sheet

Calculate the bistatic radar cross section of an electrically thin dielectric sheet. The sheet is modelled using the thin dielectric sheet approximation and is illuminated by an incident plane wave.

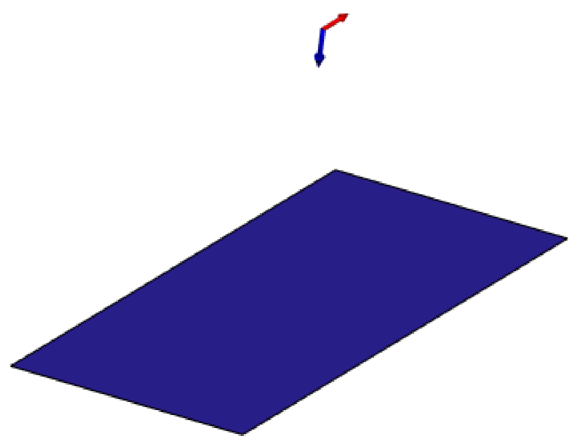


Figure 66: 3D view of a thin dielectric sheet.

C.1.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
- *freq* = 100e6 (The operating frequency.)
- *d* = 0.004 (Plate thickness.)
- *a* = 2 (Width of plate.)
- *b* = 1 (Depth of plate.)
- *epsr* = 7 (Relative permittivity.)
- *tand* = 0.03 (Loss tangent.)
- *thetai* = 20 (Zenith angle of incidence.)
- *phii* = 50 (Azimuth angle of incidence.)
- *etai* = 60 (Polarisation angle of incident wave.)

2. Create the media for the thin dielectric.
   a) Create a dielectric with the relative permittivity set to *epsr*, dielectric loss tangent to *tand*. Rename its label to `substrate`.

b) Create a single-layered dielectric.
- **Label**: `thin_dielsheet`
- **Thickness**: *d*
- **Dielectric material**: `substrate`

3. Create a rectangular plate centred at the origin in the XY plane.
   a) Create a rectangle.
   - **Definition method**: **Base centre, width, depth**
   - **Base centre**: (0, 0, 0)
   - **Width**: *a*
   - **Depth**: *b*
4. Set the face medium of the rectangular plate to **thin_dielsheet**.
5. Add a single incident plane wave source from direction θ=*thetai* and ϕ = *phii*. Set the polarisation angle to *etai*.
6. Set the frequency to *freq*.

C.1.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a vertical far field request (-180°≤θ≤180°, with ϕ=0°). Sample the far field at θ=2° steps.

C.1.3 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

C.1.4 Viewing the Results

View and post-process the results in POSTFEKO.

View the bistatic RCS of the dielectric sheet at 100 MHz as function of the angle θ, in the plane ϕ=0°.

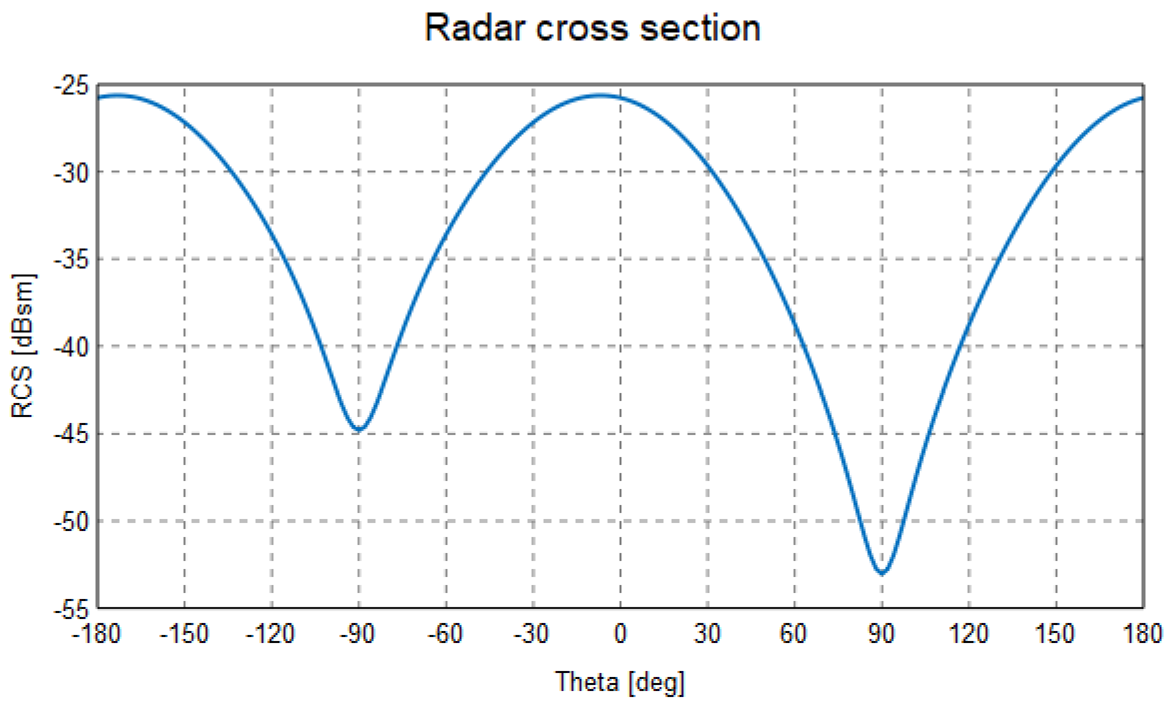


Figure 67: Bistatic RCS of a thin dielectric sheet.

C.2 RCS and Near Field of a Dielectric Sphere

Calculate the radar cross section and the near field inside and outside of a dielectric sphere using the surface equivalence principle (SEP).

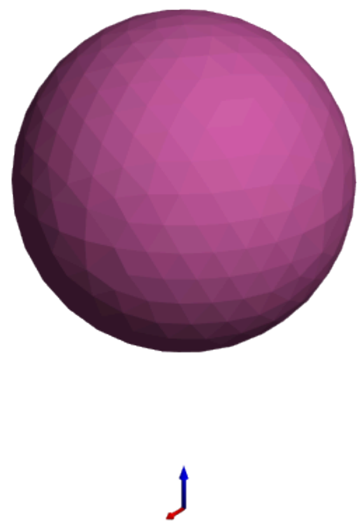


Figure 68: 3D view of a dielectric sphere with a plane wave (source).

The bistatic radar cross section for sphere[2] is computed by:

$$\sigma\left(bistatic\right) = \lim_{r\to\infty}\left[4\pi r^2 \frac{|\mathbf{E}^s|^2}{|\mathbf{E}^i|^2}\right]. \tag{4}$$

C.2.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
    - *lambda* = 20 (The wavelength in free space.)
    - *freq* = *c0*/*lambda* (The operating frequency.)
    - *radius* = 1 (Sphere radius.)
    - *epsilon* = 36 (Relative permittivity.)
2. Create a new dielectric labeled `diel` with relative permittivity set to *epsilon*.
3. Create a sphere.
    - **Centre**: (0, 0, 0)
    - **Radius**: *Radius*

2. C. A. Balanis, Advanced Engineering Electromagnetics, Wiley, 1989, pp. 655.

4. Set the region of the sphere to *diel*.
5. Add a single incident plane wave with θ=180° and ϕ=0°.
6. Set the frequency to *freq* (≈15 MHz)

C.2.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Create a vertical far field request (0°≤θ≤180°, with ϕ=0°). Sample the far field at θ=2° steps.
2. Create a near field request along the Z axis.
    a) **Definition method**: **Cartesian**
    b) Select **Specify number of points** from the list.
    c) **Start**: (0, 0, -2**radius*)
    d) **End**: (0, 0, 2**radius*)
    e) **Number of field points**: (1, 1, 80)

C.2.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Mesh size** to **Fine**.

> **Tip:** The wavelength is large compared to the size of the sphere requiring a mesh that accurately represents the geometry. Use a **Custom** mesh size for an even finer mesh.

C.2.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

C.2.5 Viewing the Results

View and post-process the results in POSTFEKO.

1. Compare the near field along the Z axis between the exact and the computed near field.

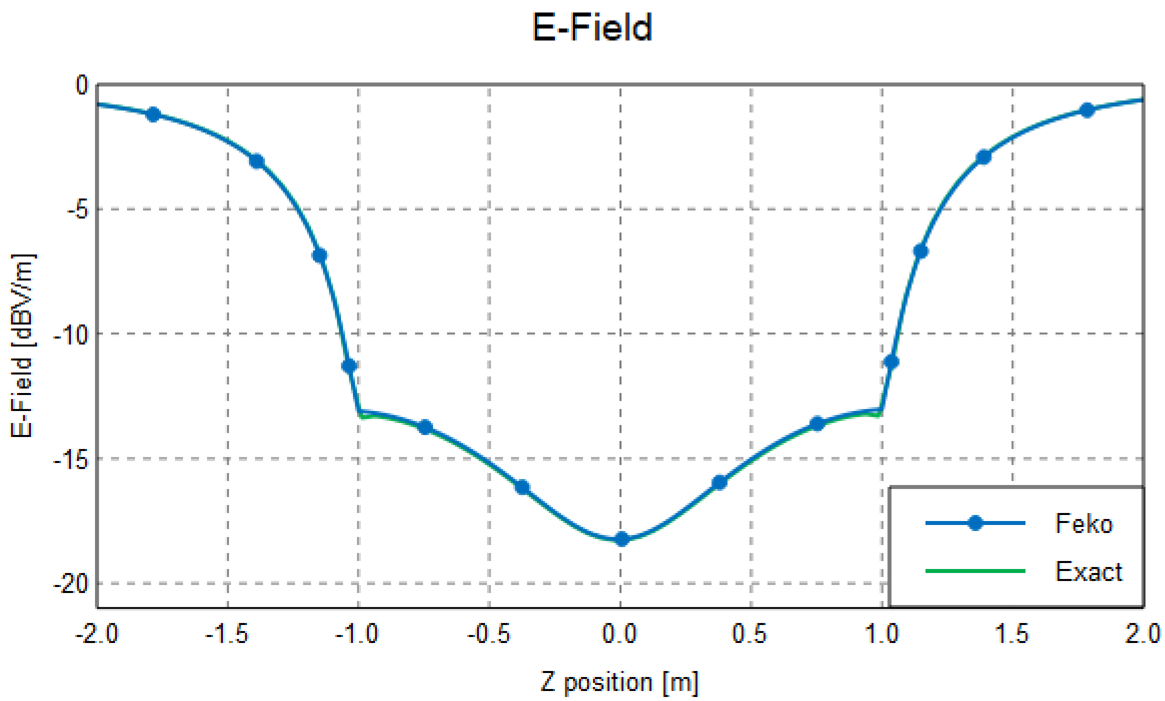


Figure 69: Near field along the Z axis.

2. Compare the RCS results on a far field graph.

a) Change the Y axis to a logarithmic scale for improved visualisation.

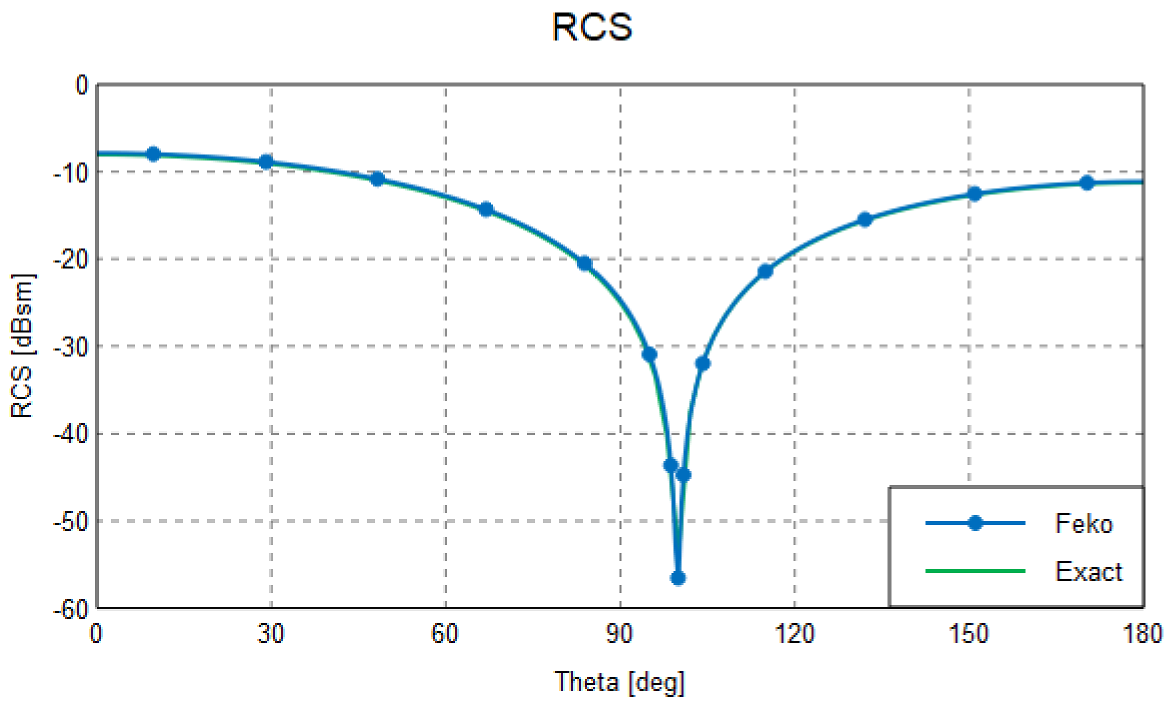


Figure 70: Bistatic radar cross section of the dielectric sphere.

Compare the results with the literature reference *(C. A. Balanis, Advanced Engineering Electromagnetics, Wiley, 1989, pp. 607.).* The results agree well with the literature reference.

C.3 Scattering Width of an Infinite Cylinder

Calculate the scattering width of an infinite cylinder. The infinite cylinder is modelled using an one-dimensional periodic boundary condition (PBC).

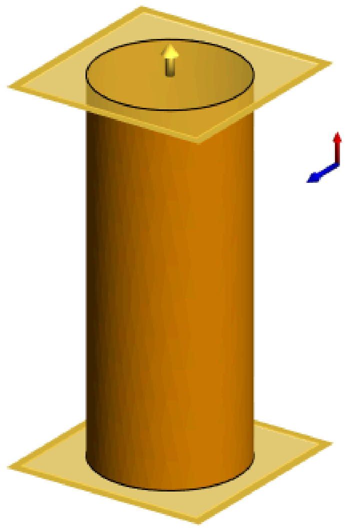


Figure 71: 3D view of the infinite cylinder defined as a unit cell with a one-dimensional PBC with a plane wave (source).

The scattering by a circular cylinder[3] is computed by:

$$SW = \frac{1}{\lambda} \lim_{\rho \to \infty} \left[2\pi\rho \frac{|E_z^s|^2}{|E_z^i|^2} \right]. \quad (5)$$

C.3.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

The model consists of a circular cylinder. The cylinder height is set to half a wavelength at the excitation frequency.

1. Define the following variables:
   - *lambda* = 1 (The wavelength in free space.)
   - *freq* = *c0*/*lambda* (The frequency for free space wavelength.)
   - *h* = *lambda*/2 (The height of the cylinder.)

3. C. A. Balanis, Advanced Engineering Electromagnetics, Wiley, 1989, pp. 607.

- *r* = 0.1 (The radius of the infinite cylinder.)

2. Create a cylinder.
    - **Definition method**: **Base centre, radius, height**
    - **Base centre**: (0, 0, -*h*/2)
    - **Radius**: *r*
    - **Height**: *h*
3. Delete the top and bottom faces of the cylinder.
4. Add a single incident plane wave with θ=90° and ϕ=180°.
5. Set the frequency to the variable *freq*.

C.3.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

This model considers the scattering width of an infinite cylinder. The incident plane wave is normal to the cylinder.

1. Define a one dimensional periodic boundary condition.
    - **Start point**: (0, 0, -*h*/2)
    - **End point of first vector**: (0, 0, *h*/2)

    > **Note:** The geometry is allowed to touch the periodic boundaries.

2. Create a near field request. The scattering width is derived from the direction-dependent scattered field.
    - **Definition method**: **Cylindrical**
    - Select **Specify increments** from the list.
    - **Start**: (500*lambda, 0, 0)
    - **End**: (500*lambda, 360, 0)
    - **Increment**: (0, 0.5, 0)
    - Calculate only the scattered part of the field. This removes the effect of the plane wave on the calculated field. As a result only the scattered fields are considered.

        > **Tip:** Open the **Request/Modify near fields** dialog, click the **Advanced** tab and then click the **Calculate only the scattered part of the field** check box.

C.3.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Create the mesh by using the **Fine** auto-mesh setting.

C.3.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

C.3.5 Viewing the Results

View and post-process the results in POSTFEKO.

View the computed scattering width as a function of the bistatic observation angle (ϕ) for a cylinder radius of *r*=0.1 and *r*=0.6.

a) The scattering width is obtained by using the equation at the top of the example with the values provided to simplify to

$$SW = 2\pi 500 |E|^2. \tag{6}$$

b) To use the equation in POSTFEKO, ensure the magnitude of the electric field is displayed.

c) Select the **Enable maths** check box and enter the following:

```
2*pi*500*ABS(self)^2
```

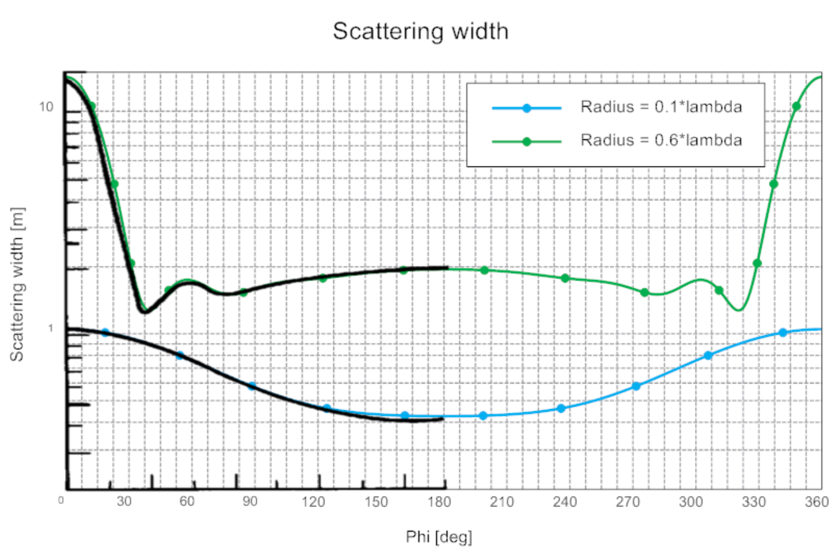


Figure 72: The scattering width of an infinite cylinder, modelled with r=0.1 and r=0.6 where r is the radius of the cylinder.

Compare the results with the literature reference *(C. A. Balanis, Advanced Engineering Electromagnetics, Wiley, 1989, pp. 607.).* The results agree well with the literature reference.

C.4 Periodic Boundary Conditions for FSS Characterisation

Calculate the transmission and reflection coefficients for a Jerusalem cross FSS (frequency selective surface) structure. The cross is modelled with a periodic boundary condition and is excited with an incident plane wave.

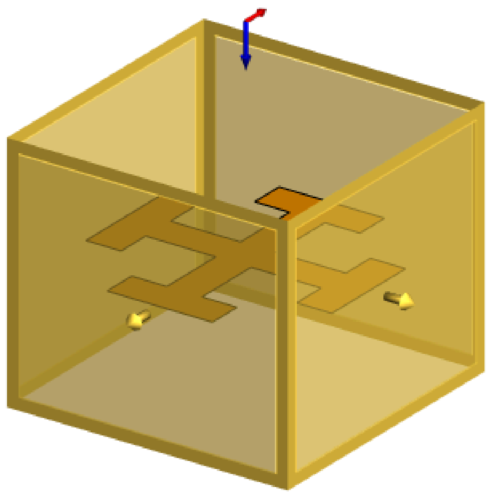


Figure 73: 3D view of the Jerusalem cross unit-cell structure defined as a unit cell with two-dimensional PBC with a plane wave (source) .

C.4.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
    - Set the model unit to millimetres (mm).
    - *d* = 15.2 (The spacing for periodic boundary condition.)
    - *fmin* = 2e9 (The minimum frequency.)
    - *fmax* = 12e9 (The maximum frequency.)
    - *armLength* = 13.3 (The arm length of the cross.)
    - *armWidth* = 1.9 (The arm width of the cross.)
    - *stubLength* = 5.7 (Stub length at the end of the cross.)
    - *stubWidth* = *armWidth* (Stub width at the end of the cross.)
2. Create the main arm of the cross.

a) Create a rectangle centred at the origin.
   - **Definition method**: **Base centre, width, depth**
   - **Base centre**: (0, 0, 0)
   - **Width**: *armWidth*
   - **Depth**: *armLength*

3. Create the stub rectangle.
   a) Create a rectangle centred at the origin.
   - **Definition method**: **Base centre, width, depth**
   - **Base centre**: (0, 0, 0)
   - **Width**: *stubLength*
   - **Depth**: *stubWidth*
4. Translate the stub rectangle as follows:
   - **From**: (0, -*stubWidth*/2, 0)
   - **To**: (0, -6.65, 0)
5. Copy and mirror the stub across the XZ plane.
   There should now be a stub at each end of the main arm.
6. Copy and rotate all the parts by 90°.
7. Union all the parts and simplify **Union1**.
8. Add a single incident plane wave with θ=0° and ϕ=0°.
9. Set a continuous frequency range from *fmin* to *fmax*.

C.4.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

This model considers the transmission and reflection coefficients for an incident plane wave.

1. Request **Transmission/reflection coefficients** with the phase origin at (0, 0, 0).
2. Define **Periodic Boundary Conditions** in **Two dimensions**.
   a) **Start point**: (-*d*/2 , -*d*/2 ,0)
   b) **End point of first vector**: (*d*/2 , -*d*/2 ,0)
   c) **End point of second vector**: (-*d*/2 , *d*/2 ,0)
   d) For **Phase shift** select **Determine from plane-wave excitation**.

C.4.3 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

C.4.4 Viewing the Results

View and post-process the results in POSTFEKO.

View the total computed transmission and reflection coefficients.

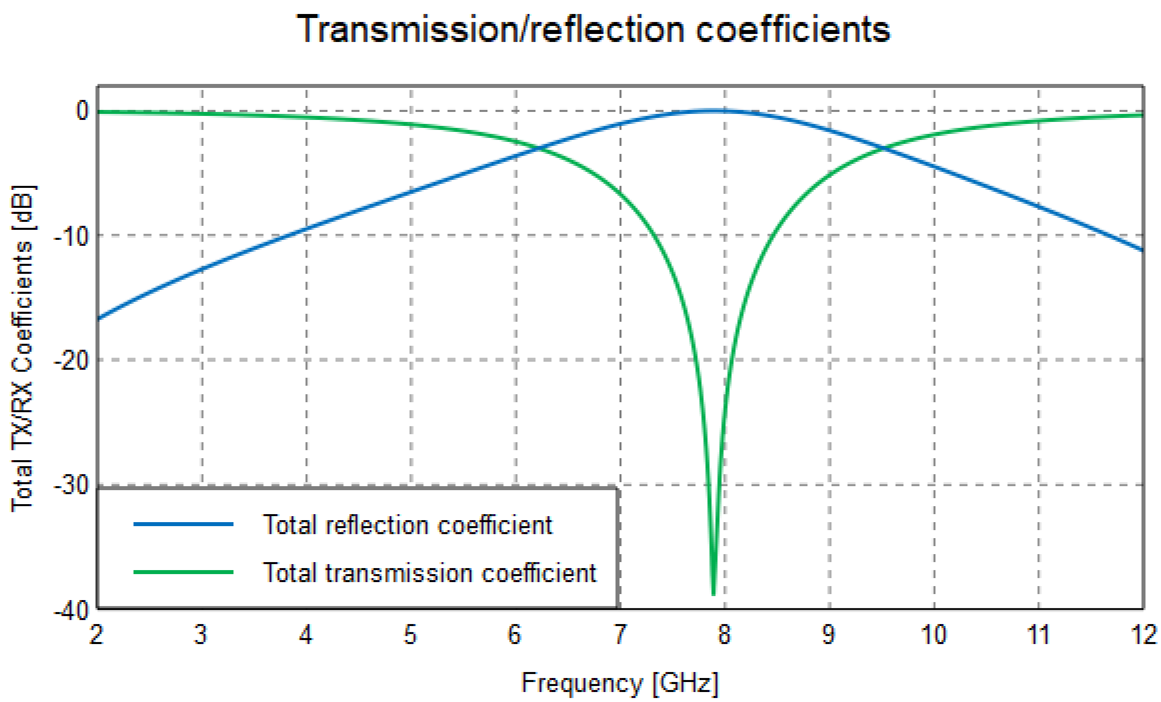


Figure 74: The transmission and reflection coefficients for the specified incident plane wave.

Compare the results with the literature reference, *Ivica Stevanovic, Pedro Crespo-Valero, Katarina Blagovic, Frederic Bongard and Juan R. Mosig, Integral-Equation Analysis of 3-D Metallic Objects Arranged in 2-D Lattices Using the Ewald Transformation, IEEE Trans. Microwave Theory and Techniques, vol. 54, no. 10, October 2006, pp. 3688–3697.*

C.5 Bandpass FSS

Calculate the transmission coefficient of a second order, electrically thin bandpass FSS (frequency selective surface) structure. The structure is modelled with a periodic boundary condition and is excited with an incident plane wave.

Model the structure using the following configurations:

1. The method of moments (MoM) solution with the surface equivalence principle (SEP).
2. The hybrid finite element method and method of moments (FEM/MoM) method.

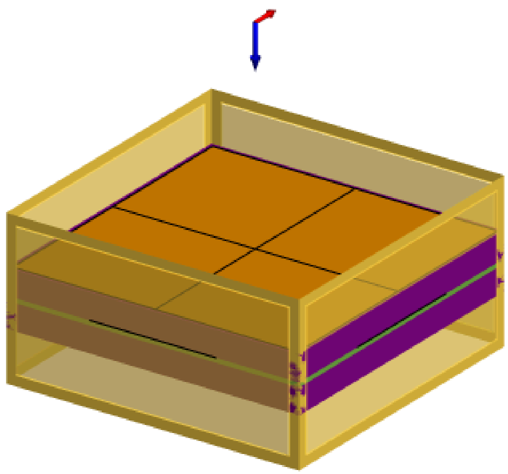


Figure 75: 3D view view of the second order FSS.

Compare the transmission coefficient with the reference.[4]

C.5.1 SEP Model

The frequency selective surface (FSS) is solved using the surface equivalence principle (SEP) method.

> **Note:** Expect several hours of computation time for the SEP model.

4. A New Technique for Design of Low-Profile Second-Order, Bandpass Frequency Selective Surfaces, Mudar Al-Joumayly and Nader Behdad, IEEE Trans. Antennas and Propagation, Vol. 57, No.2, Feb 2009

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to millimeters.
2. Define the following variables:
    - *d* = 5.8 (The spacing for the periodic boundary condition.)
    - *h* = 0.5 (The thickness of the substrate.)
    - *t* = 0.091 (The thickness of the bonding material.)
    - *s1* = 0.15 (The width of the dielectric surface, bottom layer.)
    - *s2* = 0.18 (The width of the dielectric surface, upper layer.)
    - *w* = 2.5 (The width of the centre metallic surface.)
    - *fmin* = 5e9 (The minimum frequency.)
    - *fmax* = 15e9 (The maximum frequency.)
3. Create the two layers for the dielectric.
    a) Create a cuboid for the lower layer centred at the origin.
        - **Definition method**: **Base centre, width, depth, height**
        - **Base centre**: (0, 0, 0)
        - **Width**: *d*
        - **Depth**: *d*
        - **Height**: *h*
        - **Label**: `LowerLayer`

    b) Create a cuboid for the middle layer.
        - **Definition method**: **Base centre, width, depth, height**
        - **Base centre**: (0, 0, *h*)
        - **Width**: *d*
        - **Depth**: *d*
        - **Height**: *t*
        - **Label**: `Glue`

    c) Create a cuboid for the upper layer.
        - **Definition method**: **Base centre, width, depth, height**
        - **Base centre**: (0, 0, *h* + *t*)
        - **Width**: *d*
        - **Depth**: *d*
        - **Height**: *h*
        - **Label**: `UpperLayer`
4. Create the lower and upper PEC surfaces.
    a) Create the upper rectangle.
        - **Definition method**: **Base centre, width, depth**

- **Base centre**: (0, 0, *h* + *t* + *h*)
- **Width**: *d-s1*
- **Depth**: *d-s1*
- **Label**: `S1`

b) Create the lower rectangle.

- **Definition method**: **Base centre, width, depth**
- **Base centre**: (0, 0, 0)
- **Width**: *d* - *s2*
- **Depth**: *d* - *s2*
- **Label**: `S2`

5. Create the outline of the PEC surface on the bonding material.

a) Create a rectangle.

- **Definition method**: **Base centre, width, depth**
- **Base centre**: (0, 0, *h* + *t*)
- **Width**: *d*
- **Depth**: *w*
- **Label**: `L1`

b) Create a rectangle.

- **Definition method**: **Base centre, width, depth**
- **Base centre**: (0, 0, *h* + *t*)
- **Width**: *w*
- **Depth**: *d*
- **Label**: `L2`

6. Union the parts, **L1** and **L2**.
7. Use the **Simplify** transform on **Union1** to remove redundant edges caused by the overlapping of two rectangles.
8. Union all the parts.
9. Create the two dielectric media.

a) Create the dielectric medium, `Substrate`.

- **Relative permittivity**: `3.38`
- **Dielectric loss tangent**: `0`
- **Label**: `Substrate`

b) Create the dielectric medium, `Glue`.

- **Relative permittivity**: `3.3`
- **Dielectric loss tangent**: `0`
- **Label**: `Glue`

10. Set the region properties of the two layers.

a) Set the region of the middle layer to `Glue`.

b) Set the region of the remaining two layers to `Substrate`.

11. Set the face centred on top of the *Glue* material to PEC.

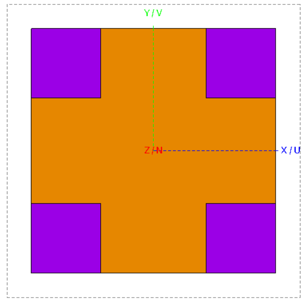


Figure 76: Top view of the FSS using show/hide of faces or a cutplane to show the PEC face centred on top of the glue material.

> **Tip:** Use show/hide of faces to see these faces.

12. Set the large top face and large bottom face to PEC, but keep the remaining narrow faces (which border the large top face and bottom face respectively) set to **Default** or **Dielectric boundary**.

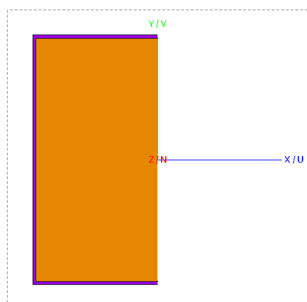


Figure 77: Top view of the FSS. The top and bottom faces are set to PEC. The narrow faces around the border are set to default or dielectric boundary.

13. Add a single incident plane wave with θ=0° and ϕ=0°.

14. Set a continuous frequency range from *fmin* to *fmax*.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

This model considers the transmission coefficients for an incident plane wave.

1. Request **Transmission/reflection coefficients** with the phase origin at (0, 0, *h*+*t*+*h*).
2. Define a 2D periodic boundary condition.
   a) **Start point**: (*d*/2 , *d*/2 ,0)
   b) **End point of first vector**: (-*d*/2 , *d*/2 ,0)

c) **End point of second vector**: (*d*/2 , -*d*/2 ,0)

d) For **Phase shift** select **Determine from plane-wave excitation**.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Specify local mesh refinement.
   a) Set a local mesh size of *d*/50 on the dielectric surfaces of the narrow faces (which border the large top face and bottom face respectively)[5].

   > **Tip:** Open the **Modify Face** dialog, click the **Meshing** tab and then select the **Local mesh size** check box.

   a) Set a local mesh size of *d*/50 on the *Glue* region.
2. Set the **Mesh size** equal to **Fine**.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

5. See Figure 77.

C.5.2 Hybrid FEM / MoM Model

The frequency selective surface (FSS) is solved using the hybrid finite element method and method of moments (FEM /MoM) solution method.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in SEP Model and rename the file.
2. Set the solution method for all regions to FEM.

   > **Tip:** Open the **Modify Region** dialog and click the **Solution** tab. From the **Solution Method** drop-down list, select **Finite Element Method (FEM)**.

3. Set the data storage precision to **Double precision** for faster convergence.

   > **Note:** Open the **Solver settings** dialog and click the **General** tab. Under **Data storage precision**, select **Double precision**.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Define the same calculation requests as for the SEP Model.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for the SEP Model.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

C.5.3 Viewing the Results

View and post-process the results in POSTFEKO.

1. View the total computed transmission coefficient of both the SEP and FEM/MoM on a Cartesian graph.

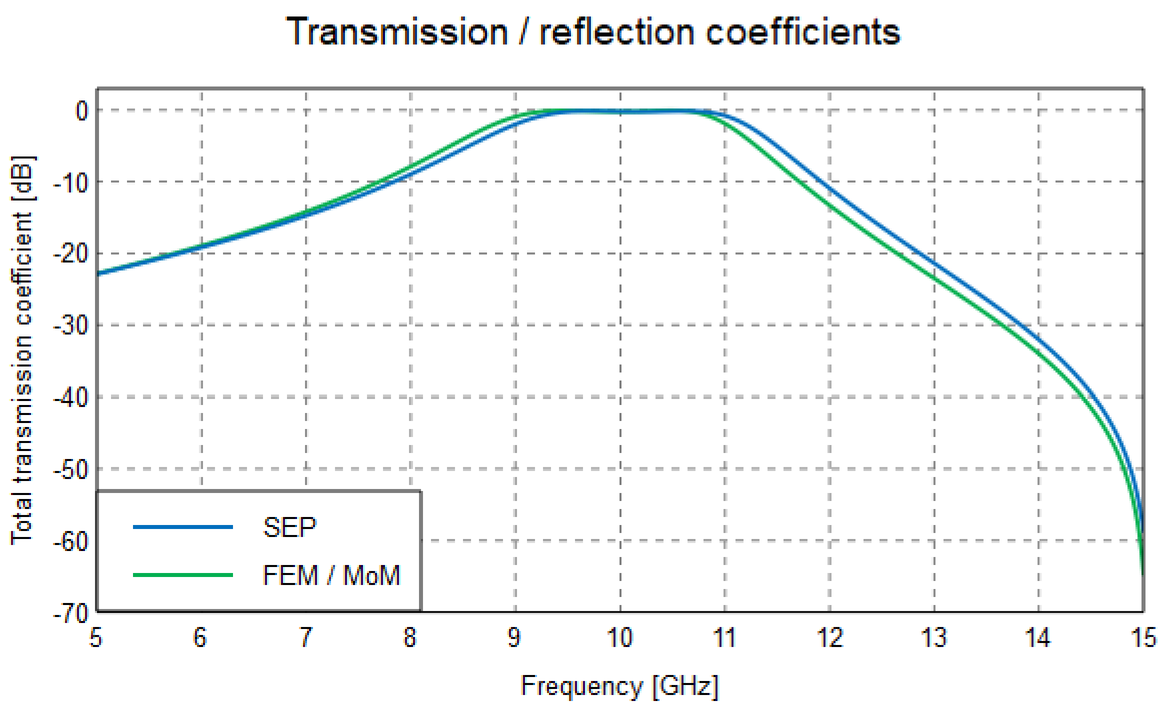


Figure 78: The transmission coefficients for both the SEP and FEM / MoM solution methods.

The FEM is a computationally more efficient solution.

2. [Optional] Compare the results with the literature reference *A New Technique for Design of Low-Profile Second-Order, Bandpass Frequency Selective Surfaces, Mudar Al-Joumayly and Nader Behdad, IEEE Trans. Antennas and Propagation, Vol. 57, No.2, Feb 2009*.

EMC Analysis and Cable Coupling

D

Simple examples demonstrating electromagnetic compatibility (EMC) analysis and cable coupling.

This chapter covers the following:

D.1 Shielding Factor of a Sphere with Finite Conductivity

Calculate the shielding factor of a hollow sphere with finite conductivity. The sphere is constructed from a lossy metal with a thickness of 2.5 nm.

An incident plane wave is defined from 1 MHz to 100 MHz. Use a single near field point located at the centre of the sphere to compute the shielding factor.

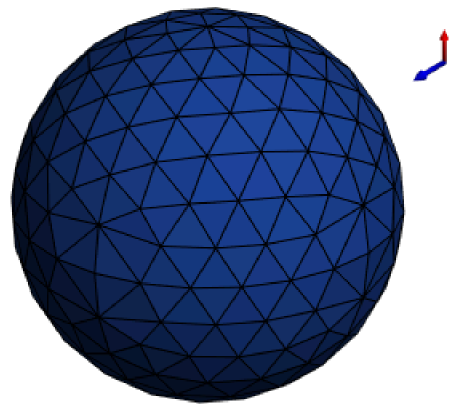


Figure 79: A 3D view of the sphere with a plane wave (source) and symmetry in CADFEKO.

The shielding factor results are compared using the following solution methods:

- Method of moments (MoM)
- Finite element method (FEM)

> **Tip:** Each model uses its predecessor as a starting point. Create the models in their presentation order. Save each model to a new location to keep them.

D.1.1 MoM Model

Calculate the shielding factor of a hollow sphere with finite conductivity. The sphere is solved using MoM.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
    - *r0* = 1 (Sphere radius.)
    - *f_min* = 1e6 (The minimum frequency.)
    - *f_max* = 100e6 (The maximum frequency.)
    - *d* = 2.5e-9 (Thickness of the shell.)
2. From the media library, add the predefined medium, **Silver** to the model.
3. Create a sphere.
    - **Definition method**: **Centre, radius**
    - **Centre**: (0, 0, 0)
    - **Radius**: *r0*
    - **Label**: `Sphere1`
4. Set the region of the sphere to **Free space**.
5. Set the face of the sphere to the medium, **Silver**. Set the thickness to *d*.
6. Create a single incident plane wave source with θ=90° and ϕ=180°.
7. Set a continuous frequency range from *f_min* to *f_max*.
8. Specify the symmetry about 3 principal planes:
    a) X=0: **Geometric symmetry**
    b) Y=0: **Magnetic symmetry**
    c) Z=0: **Electric symmetry**

> **Tip:** Exploit model symmetries (if it exists) in a large or complex model to reduce computational costs.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a near field request. The request is a single point located at the centre of the sphere.

- **Definition method**: **Cartesian**
- Select **Specify number of points** from the list.
- **Start**: (0, 0, 0)
- **End**: (0, 0, 0)

- **Number of field points**: (1, 1, 1)

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Mesh size** equal to **Standard**.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

D.1.2 FEM Model

Calculate the shielding factor of a hollow sphere with a finite conductivity. The sphere is modelled with FEM.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in MoM Model and rename the file.
2. Define the following variable:
   - *r1* = 1.2 (Radius of FEM vacuum sphere.)
3. Create a new dielectric labeled `air` with the default properties of **Free space**.
4. Create a sphere.
   - **Definition method**: **Centre, radius**
   - **Centre**: (0, 0, 0)
   - **Radius**: *r1*
5. Set the regions of both spheres to `air`.

   > **Tip:** A dielectric with similar properties to free space is used instead of free space. It allows the region to be meshed as a tetrahedral volume for FEM.

6. Union the two spheres.
7. Set the solver method for the regions to FEM.
8. Add a single incident plane wave with θ=90° and ϕ=180°.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for MoM Model.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Use the same mesh settings as for MoM Model.
2. Mesh the model.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

D.1.3 Viewing the Results

Compare the shielding factor results for the sphere using the MoM and FEM solution methods.

View the shielding factor of the sphere with respect to the incident electric and magnetic fields.

> **Tip:** Calculate the ratio between the field measured inside the sphere and the field incident on the sphere.

a) Incident electric field on sphere: The incident electric field strength was set to E_i = 1 V/m.

b) Incident magnetic field on sphere: Calculate the incident magnetic field from the wave impedance for a plane wave as follows:

$$H_i = \frac{E_i}{\eta_0} = \frac{1}{376.7} = 2.6544 \times 10^{-3} \quad A/m \tag{7}$$

c) The shielding factor is calculated from as follows:

$$S_e = -20 * log \frac{E}{E_i} \quad [dB] \tag{8}$$

$$S_h = -20 * log \frac{H}{H_i} \quad [dB] \tag{9}$$

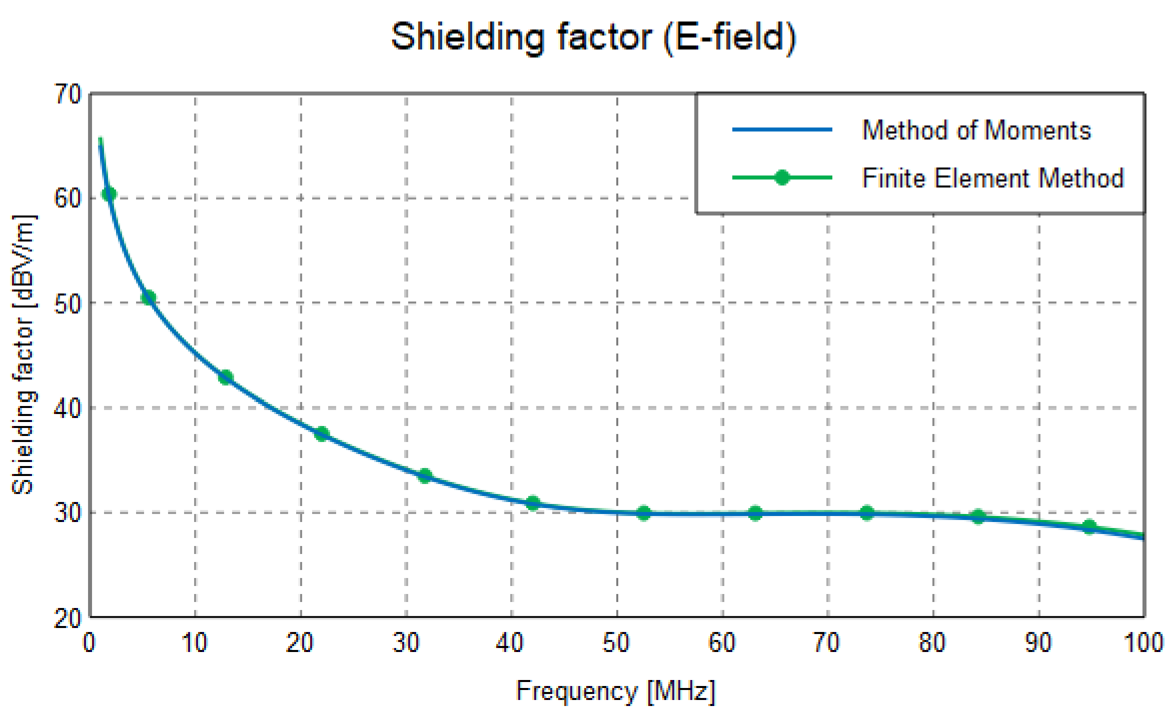


Figure 80: The shielding factor for the electric fields for a sphere with finite conductivity.

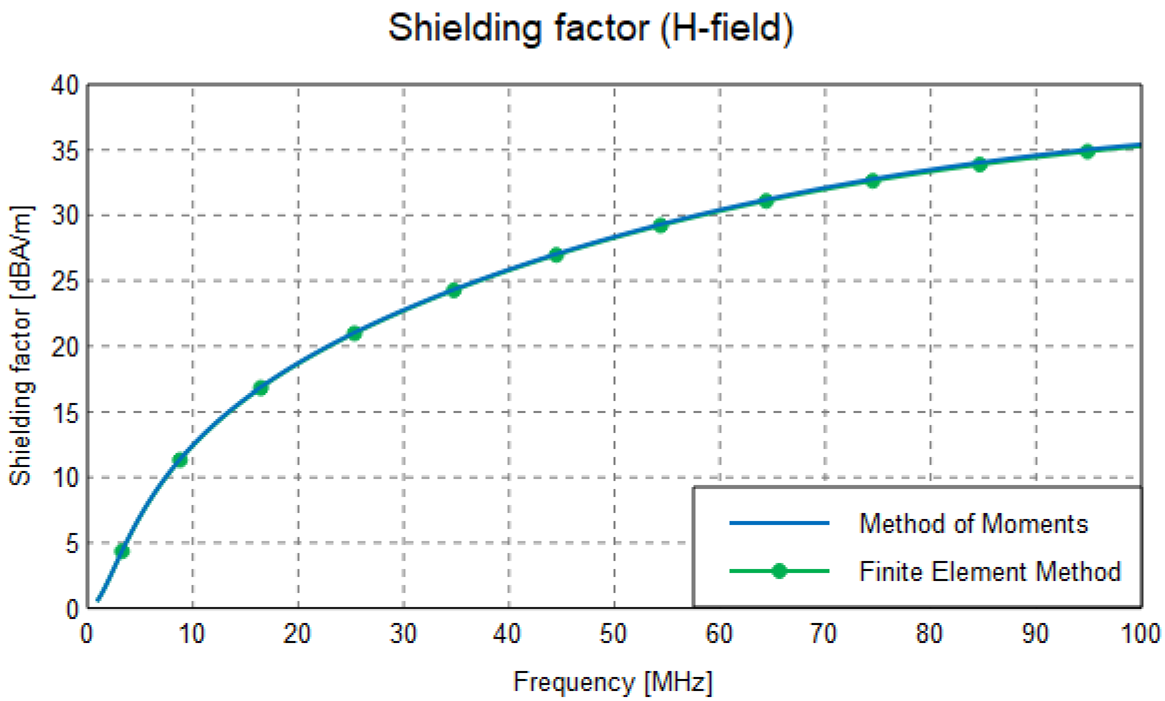


Figure 81: The shielding factor for the magnetic fields for a sphere with finite conductivity.

D.2 Calculating Field Coupling into a Shielded Cable

Calculate the coupling between a monopole antenna and a nearby shielded cable that follows an arbitrary path above a ground plane.

The cable analysis method is used to analyse the cable harness. This method first solves the model without the cable and then calculates the coupling into the cable using the cable transfer impedance. Included in the cable analysis method is a database of measured cable properties (integrated in Feko).

> **Note:** Simple cables such as unshielded cables or twisted pairs can also be modelled using the full MoM solver method. However, the memory requirement and solution time will be much larger than the cable analysis method.

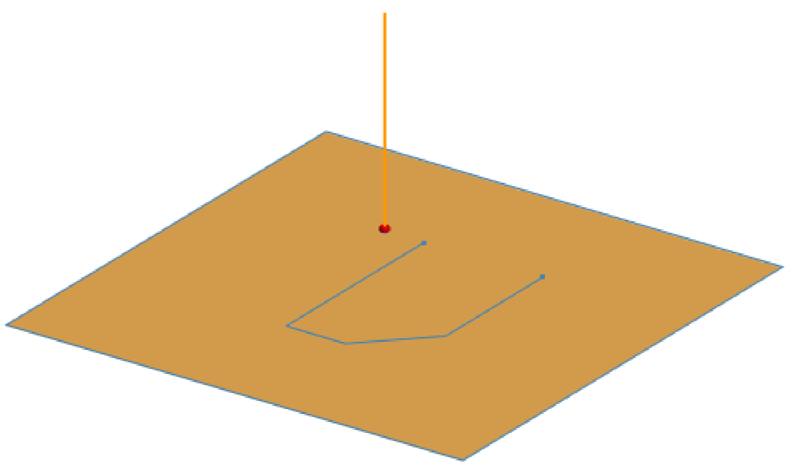


Figure 82: 3D view of an RG58 shielded cable illuminated by a monopole above an infinite ground plane.

D.2.1 Creating the Monopole and Ground Plane

Create a monopole antenna and infinite PEC ground plane.

1. Define the following variables:
    - *fmin* = 1e6 (The minimum frequency.)
    - *fmax* = 35e6 (The maximum frequency.)
    - *wireRadius* = 1e-3 (The wire radius of the monopole.)
2. Create the monopole.
    a) Create a line.
        - **Start point**: (0, 0, 0)

- **End point**: (0, 0, 10)

3. Add a wire port (segment) to the base of the line.
4. Add a voltage source to the port. (1 V, 0°, 50 Ω).
5. Set the total source power (no mismatch) to 10 W.
6. Add an infinite PEC ground plane.

D.2.2 Creating the Shielded Cable, Connections and Terminations

Create a cable path section with cable harness. Create the connections and terminations in the schematic view. Set the frequency.

1. Create a cable path with the following (x, y, z) corners:
   1. **Corner 1**: (0, 2, 0.01)
   2. **Corner 2**: (10, 2, 0.01)
   3. **Corner 3**: (10, 5, 0.01)
   4. **Corner 4:** (7, 8, 0.01)
   5. **Corner 5**: (0, 8, 0.01)
2. Create a cable harness.

   A typical harness consists of multiple cables routed along the same cable path.
3. Create two cable connectors for both ends of the cable path. Rename their labels to `startConnector` and `endConnector`.

   Both connectors will have two pins; one that is live and one for ground (cable shield).
4. Create a coaxial cable and select **RG58 C/U** from the list of predefined coaxial cable types.
5. Create a cable instance that runs from *startConnector* to *endConnector*.
   1. Connect the two live pins. Ensure the live pins connect to the centre conducting wire.
   2. Connect the two ground pins. Ensure the ground pins connect to the outer shielding of the cable.
   3. Verify that the connections are correct by looking at the labels in the preview.
6. Open the **CableHarness1** schematic view.

   a) Add a 50 Ω complex load to each connector to terminate the cable. The load must be connected between the live and ground pins.

   b) Connect the outer shields of the cables (ground pins) to the global ground in the schematic view.
7. Set a continuous frequency range from *fmin* to *fmax*.

D.2.3 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Add a voltage probe over the load termination at the connector with label *startConnector*.

Tip: If the port impedance or power is of interest, a current probe must be requested in series to the terminating load. The values can then be derived using Ohm's law.

D.2.4 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to `wireRadius`.

D.2.5 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

D.2.6 Viewing the Results

View and post-process the results in POSTFEKO.

View the voltage over the terminating load on a Cartesian graph.

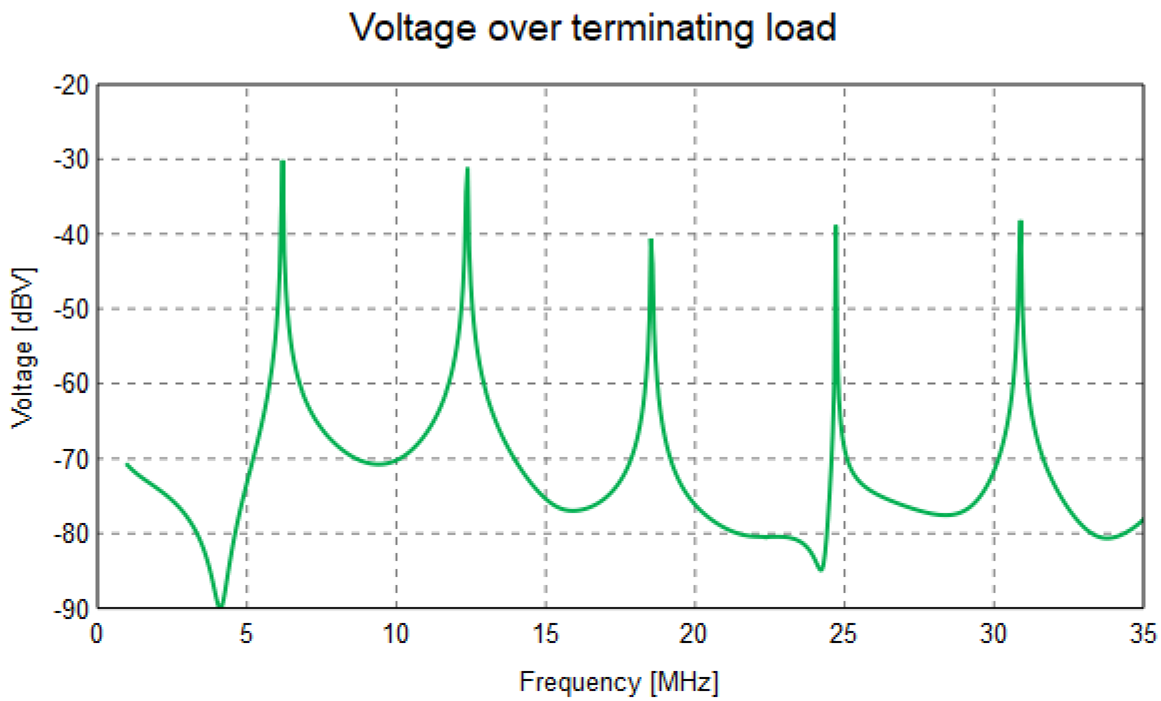


Figure 83: Voltage induced in a terminated shielded cable by an external source.

D.3 A Magnetic-Field Probe

Calculate the segment current on a magnetic-field probe as a function of the plane wave incidence angle.

The wavelength, λ, is approximately 10 m at 30 MHz.

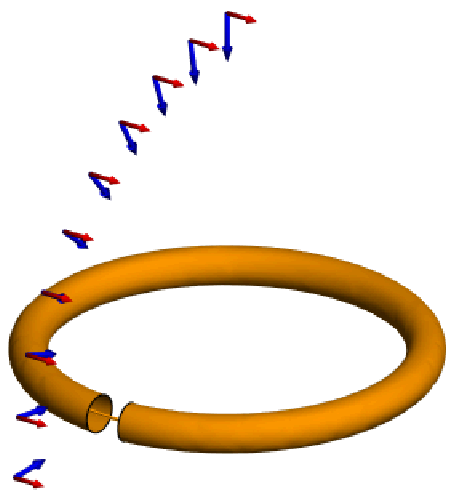


Figure 84: 3D view of the magnetic-field probe with plane wave incidence over multiple angles.

D.3.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
   - *freq* = 30e6 (The operating frequency.)
   - *lambda* = *c0*/*freq* (The wavelength in free space.)
   - *rBig* = 1 (Radius of revolution.)
   - *rSmall* = 0.1 (The pipe radius.)
   - *wireRad* = 5e-3 (The wire segment radius.)
2. Create an elliptic arc with radius set to *rSmall*. Change the workplane origin to (-*rBig* ,0,0). Set the U vector to (0, 0, 1) and V vector to (1, 0, 0).
3. Rotate the arc over an angle of 185° around the Z axis.
4. Spin the ellipse over an angle of 350° around the Z axis.
5. Draw an elliptic arc through the centre of the toroidal section. (**radius** = *rBig*, **start angle** = `0`°, **end angle** = `360`°)

6. Add an incident plane wave that loops over multiple incident angles with 0°≤θ≤90° and ϕ=0°. Set the **Polarisation (angle)** to 90°. Increment the incident angle, θ, in 10° steps.
7. Set the frequency to *freq*.

D.3.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a currents request (segment currents).

D.3.3 Meshing the Model

Create the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Create the mesh by using either the **Fine** or **Standard** auto-mesh setting.
2. Set the **Wire segment radius** equal to *wireRad*.

> **Tip:** Experiment with the advanced mesh settings. Open the **Modify mesh** dialog and click the **Advanced** tab.

D.3.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

D.3.5 Viewing the Results

Use POSTFEKO to view the results.

Create a Cartesian graph and add the segment currents to the graph.

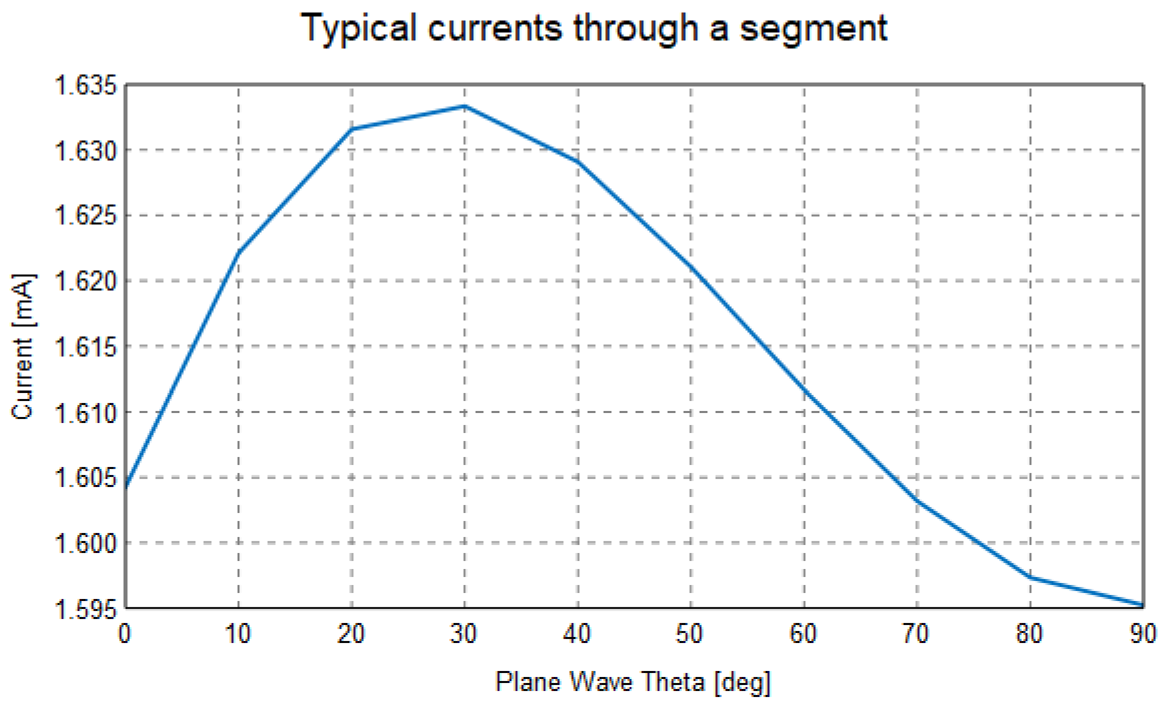


Figure 85: The current in an arbitrary segment vs plane wave source incidence angle.

Note: Each segment will result in a slightly different current as a function of the plane wave source angle of incidence.

D.4 Antenna Radiation Hazard (RADHAZ) Safety Zones

Calculate the safety zones around a Yagi-Uda antenna based on radiation INIRC88 and NRPB89 standards. View the safety zone ISO surfaces.

Calculate a full 3D near field cube of the antenna's immediate surroundings. Use math scripts to identify the safety zones.

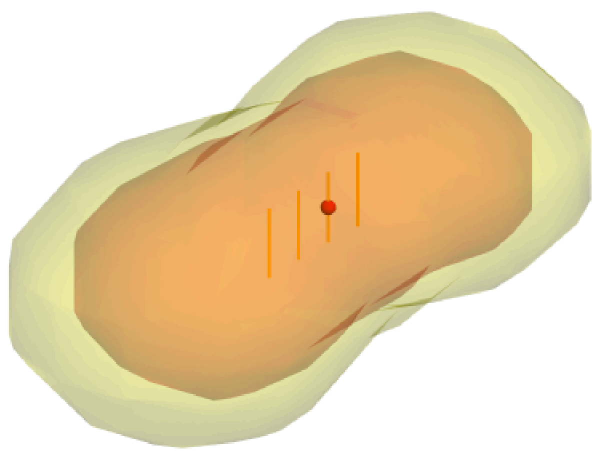


Figure 86: The 3D safety zones for 80% and maximum exposure levels according to the INIRC 88 standard at 0.95 GHz.

> **Note:** The INIRC (International Non-ionising Radiation Committee) and NRPB (The UK National Radiological Protection Board) provide standards that determine safe radiation thresholds. These standards are typically frequency dependent and defined in a piece-wise manner.

D.4.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables.
   - *freq* = 1e9 (The operating frequency.)
   - *fmin* = 0.4e9 (The minimum frequency.)
   - *fmax* = 1.5e9 (The maximum frequency.)

- *lambda* = *c0*/*freq* (The frequency for free space wavelength.)
- *L0* = 0.2375 (Length of the reflector element in wavelengths.)
- *L1* = 0.2265 (Length of the driver element in wavelengths.)
- *L2* = 0.223 (Length of the first director element in wavelengths.)
- *L3* = 0.223 (Width of the second director element in wavelengths.)
- *S0* = 0.3 (Spacing between the reflector and driver element in wavelengths.)
- *S1* = 0.3 (Spacing between the driver element and first director element.)
- *S2* = 0.3 (Spacing between directors.)
- *r* = 0.1e-3 (Wire radius in mm.)

2. Create the dipole (driven element) of the Yagi-Uda antenna.
   a) Create a line.
      - **Start point**: (0, 0, -*L1**lambda*)
      - **End point**: (0, 0, *L1**lambda*)
   b) Add a wire port to the middle of the line.
   c) Add a voltage source to the port. (1 V, 0°, 50 Ω).
3. Create the reflector of the Yagi-Uda antenna.
   a) Create a line.
      - **Start point**: (-*S0**lambda*, 0, -*L0**lambda*)
      - **End point**: (-*S0**lambda*, 0, *L0**lambda*)
4. Create the first director of the Yagi-Uda antenna.
   a) Create a line.
      - **Start point**: (*S1**lambda*, 0, -*L2**lambda*)
      - **End point**: (*S1**lambda*, 0, *L2* *lambda*)
5. Create the second director of the Yagi-Uda antenna.
   a) Create a line.
      - **Start point**: ((*S1*+*S2*)*lambda*, 0, -*L3**lambda*)
      - **End point**: ((*S1* + *S2*)*lambda*, 0, *L3**lambda*)
6. Set the incident power as follows:
   a) Select **Incident power (transmission line model)**.
   b) **Source power (Watt)**: 25
   c) **Real part of Z0**: 50
7. Set the continuous frequency range from *fmin* to *fmax*.

D.4.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a near field request.

- **Definition method**: **Cartesian**

- Select **Specify number of points** from the list.
- **Start**: `(-0.6, -0.6, -0.6)`
- **End**: `(1.2, 0.6, 0.6)`
- **Number of field points**: (20, 10, 10)

D.4.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to *r*.

D.4.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

D.4.5 Viewing the Results

View and post-process the results in POSTFEKO.

> **Note:** The session, `radiation_zones.pfs`, is included in the Feko installation.

The session contains the results from the antenna simulation and the following three scripts:

- *INIRC88*
  This script generates a near field result that incorporates the calculated near fields and the INIRC 88 safety standards for occupational limits.

  NRPB89
  This script generates a near field result that incorporates the calculated near fields and the NRPB 89 safety standards.

  standards
  This is a custom dataset that contains both the INIRC 88 and NRPB 89 safety standards. The result shows the maximum field limits for both magnetic and electric fields over the calculated frequency band.

1. View the electric and magnetic field limits for the INIRC 88 and NRPB 89 safety standards over the frequency band.

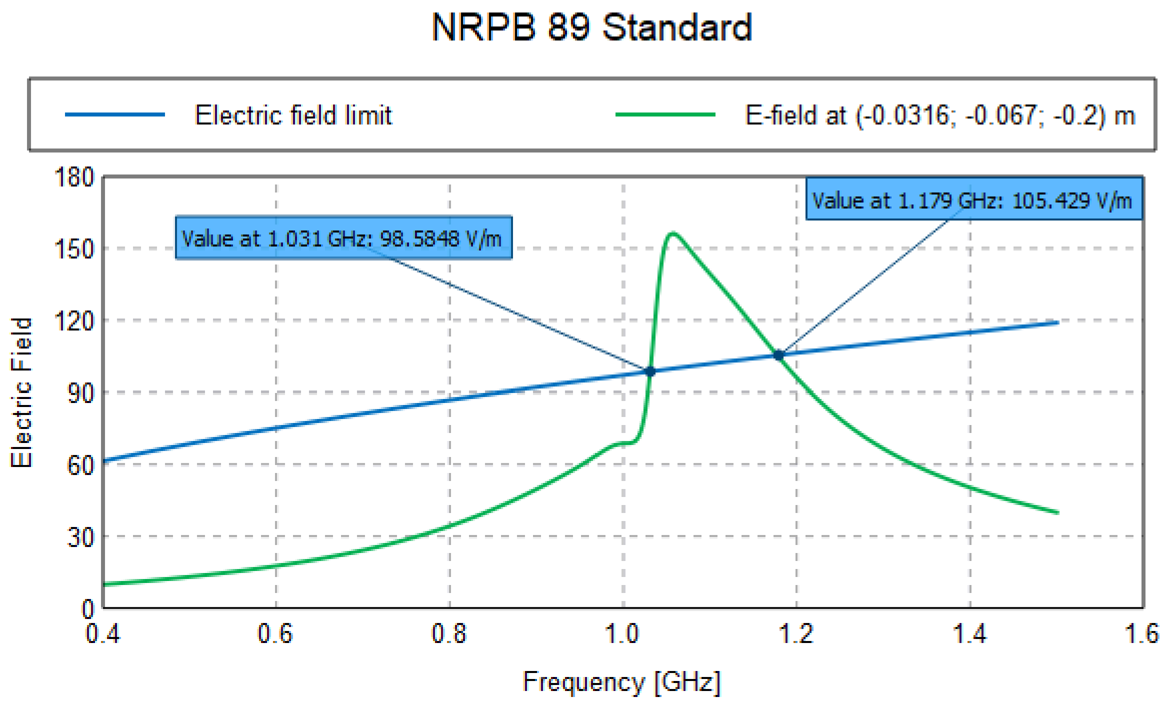
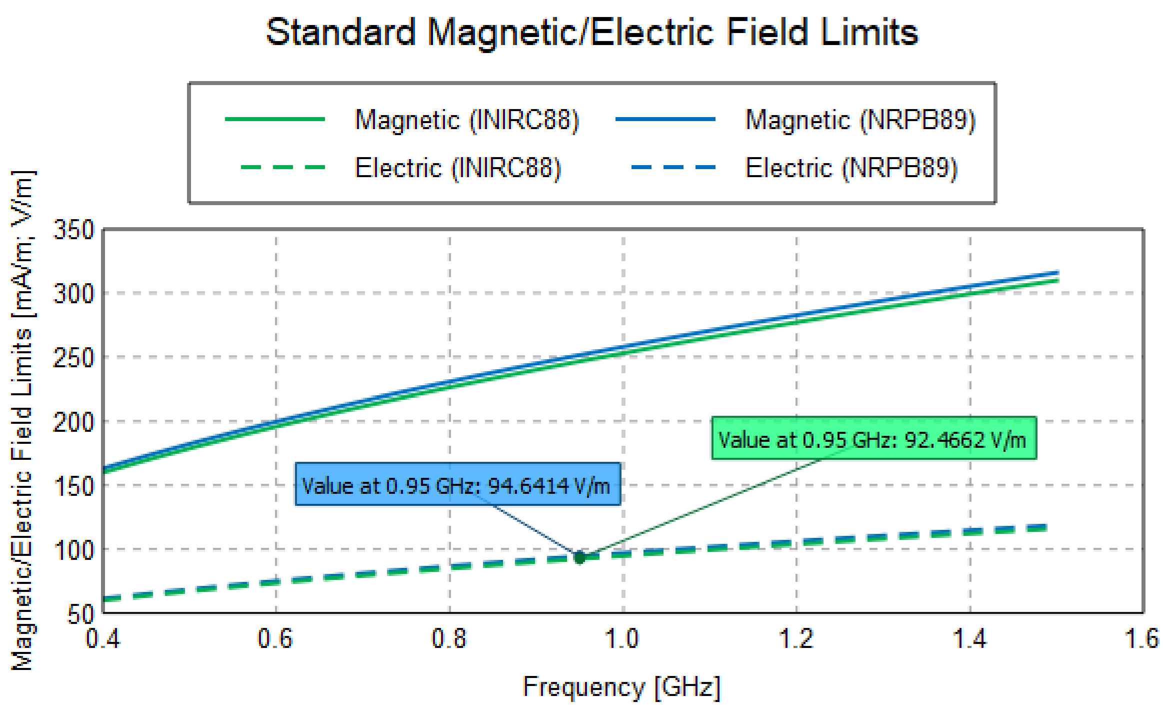


Figure 87: The electric field values at a given location (-0.0316, -0.067, -0.2) m over frequency.

2. Determine if the point (-0.0316, -0.067, -0.2) m is within the safety limits.

Figure 88: The electric and magnetic field limits for the INIRC 88 and NRPB 89 safety standards.



> **Note:** The electric field exceeds the maximum limit over the bandwidth of 1.031 – 1.179 GHz.

Script - Create Dataset

This script illustrates how the INIRC 88 safety standards and NRPB 89 safety standards are added as a dataset that allows you to plot the threshold values on a graph.

```
-- Create a dataset containing the standards formulae for reference
standards = pf.DataSet.New()
standards.Axes:Add( pf.Enums.DataSetAxisEnum.Frequency, pf.Enums.FrequencyUnitEnum.Hz,
400e6, 1.5e9 ,21 )
standards.Quantities:Add( "E_inirc88",pf.Enums.DataSetQuantityTypeEnum.Scalar,"V/m")
standards.Quantities:Add( "E_nrpb89",pf.Enums.DataSetQuantityTypeEnum.Scalar,"V/m")
standards.Quantities:Add( "H_inirc88",pf.Enums.DataSetQuantityTypeEnum.Scalar,"A/m")
standards.Quantities:Add( "H_nrpb89",pf.Enums.DataSetQuantityTypeEnum.Scalar,"A/m")
for freqIndex = 1,standards.Axes[pf.Enums.DataSetAxisEnum.Frequency].Count do
local freqHz = standards[freqIndex]:AxisValue(pf.Enums.DataSetAxisEnum.Frequency)
local freqMHz = freqHz/1e6 -- frequency in MHz
local freqGHz = freqHz/1e9 -- frequency in GHz
local standardsPt = standards[freqIndex]
-- Electric field limits
standardsPt.E_inirc88 = 3*math.sqrt(freqMHz)
standardsPt.E_nrpb89 = 97.1*math.sqrt(freqGHz)
-- Magnetic field limits
standardsPt.H_inirc88 = 0.008*math.sqrt(freqMHz)
standardsPt.H_nrpb89 = 0.258*math.sqrt(freqGHz)
end
return standards
```

INIRC 88 Standard

The definition for electric and magnetic field limits according to INIRC 88 between 0.4 GHz to 2.0 GHz.

Field type	Definition (f in MHz)	Unit
Electric field	$$3\sqrt{f} \quad (10)$$	V/m
Magnetic field	$$0.008\sqrt{f} \quad (11)$$	A/m

Script - RADHAZ Safety Zone Using INIRC 88 Standard

This script generates a near field result that incorporates the calculated near fields and the INIRC 88 safety standards for occupational limits.

The normalised threshold as per INIRC 88

Each near field value is processed and normalised to the maximum field value that the standard allows for that frequency. A value of “1” corresponds to the field threshold according to the standard. A value higher than “1” is over the limit and a value lower than“1” is a safe zone.

```
-- This example illustrates how advanced calculations
-- can be performed to display radiation hazard zones.
-- The INIRC 88 standards are used.
nf = pf.NearField.GetDataSet("yagi.StandardConfiguration1.nf3D")
function calculateRADHAZThresholds(index, nf)
-- Get a handle on the indexed near field point
local nfPt = nf[index]
-- Set up the threshold according to the standards
-- Frequency in MHz
local freq = nfPt:AxisValue(pf.Enums.DataSetAxisEnum.Frequency)/1e6
local EfieldLimit = 3*math.sqrt(freq)
local HfieldLimit = 0.008*math.sqrt(freq)
-- SCALE THE ELECTRIC FIELD VALUES
-- Scale the values to indicate percentages. The percentage represents
-- the field value relative to the limit of the standard.
nfPt.EFieldComp1 = nfPt.EFieldComp1/(EfieldLimit)
nfPt.EFieldComp2 = nfPt.EFieldComp2/(EfieldLimit)
nfPt.EFieldComp3 = nfPt.EFieldComp3/(EfieldLimit)
-- SCALE THE MAGNETIC FIELD VALUES
-- Scale the values to indicate percentages. The percentage represents
-- the field value relative to the limit of the standard.
nfPt.HFieldComp1 = nfPt.HFieldComp1/(HfieldLimit)
nfPt.HFieldComp2 = nfPt.HFieldComp2/(HfieldLimit)
nfPt.HFieldComp3 = nfPt.HFieldComp3/(HfieldLimit)
end
pf.DataSet.ForAllValues(calculateRADHAZThresholds, nf)
-- Note that in essence, the values being returned are
-- no longer near fields. As such, interpret them
-- carefully in POSTFEKO.
return nf
```

NRPB 89 Standard

The definition for electric and magnetic field limits according to NRPB 89 between 0.4 GHz to 2.0 GHz.

Field type	Definition (F in MHz)	Unit
Electric field	$97.1\sqrt{F}$ *(12)*	V/m
Magnetic field	$0.258\sqrt{F}$ *(13)*	A/m

Script - RADHAZ Safety Zone Using NRPB 89 Standard

This script generates a near field result that incorporates the calculated near fields and the NRPB 89 safety standards.

The normalised threshold as per NRPB 89

Each near field value is processed and normalised to the maximum field value that the standard allows for that frequency. A value of "1" corresponds to the field threshold according to the standard. A value higher than "1" is over the limit and a value lower than"1" is a safe zone.

```
-- This example illustrates how advanced calculations
-- can be performed to display radiation hazard zones.
-- The NRPB 89 standards are used.
nf = pf.NearField.GetDataSet("yagi.StandardConfiguration1.nf3D")
function calculateRADHAZThresholds(index, nf)
-- Get a handle on the indexed near field point
local nfPt = nf[index]
-- Set up the threshold according to the standards
-- Frequency in GHz
local freq = nfPt:AxisValue(pf.Enums.DataSetAxisEnum.Frequency)/1e9
local EfieldLimit = 97.1*math.sqrt(freq)
local HfieldLimit = 0.258*math.sqrt(freq)
-- SCALE THE ELECTRIC FIELD VALUES
-- Scale the values to indicate percentages. The percentage represents
-- the field value relative to the limit of the standard.
nfPt.EFieldComp1 = nfPt.EFieldComp1/(EfieldLimit)
nfPt.EFieldComp2 = nfPt.EFieldComp2/(EfieldLimit)
nfPt.EFieldComp3 = nfPt.EFieldComp3/(EfieldLimit)
-- SCALE THE MAGNETIC FIELD VALUES
-- Scale the values to indicate percentages. The percentage represents
-- the field value relative to the limit of the standard.
nfPt.HFieldComp1 = nfPt.HFieldComp1/(HfieldLimit)
nfPt.HFieldComp2 = nfPt.HFieldComp2/(HfieldLimit)
nfPt.HFieldComp3 = nfPt.HFieldComp3/(HfieldLimit)
end
pf.DataSet.ForAllValues(calculateRADHAZThresholds, nf)
-- Note that in essence, the values being returned are
-- no longer near fields. As such, interpret them
-- carefully in POSTFEKO.
return nf
```

Waveguide and Microwave Circuits

E

Simple examples demonstrating using waveguides and microwave circuits.

This chapter covers the following:

E.1 Microstrip Filter

Calculate the S-parameters of a simple microstrip notch filter. Use different solvers and compare the results.

The S-parameter results are compared for the following solvers:

- Finite substrate with FEM method.
- Finite substrate with SEP method.
- Infinite substrate with planar mutilayer substrate (Green's functions).

> **Note:** Reference:
>
> G. V. Eleftheriades and J. R. Mosig, "On the Network Characterization of Planar Passive Circuits Using the Method of Moments", IEEE Trans. MTT, vol. 44, no. 3, March 1996, pp. 438-445, Figs 7 and 9.

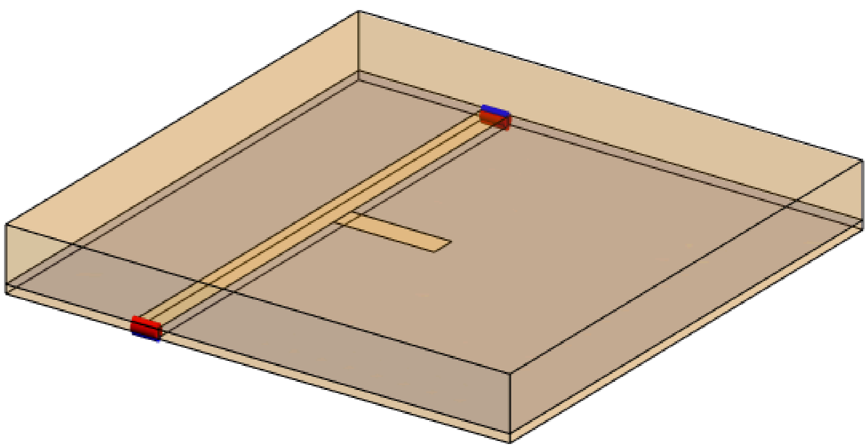


Figure 89: 3D view of the microstrip filter (cut plane view).

E.1.1 Microstrip Filter on a Finite Substrate (FEM)

Model the microstrip filter using the finite element method. Use a FEM modal port for the source.

Construct the substrate and shielding box from cuboids. Construct the microstrip line with a cuboid and delete the undesired faces. Construct the stub from a line, swept to form a surface.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to millimetres.
2. Define the following variables:
    - *shielding_height* = `11.4` (height of the shielding box.)
    - *substrate_height* = `1.57` (substrate height.)
    - *epsr* = `2.33` (relative permittivity.)
    - *gnd_length* = `92` (length and width of substrate.)
    - *port_offset* = `0.5` (inset of feed point.)
    - *strip_width* = `4.6` (width of microstrip sections.)
    - *strip_offset* = `23` (microstrip offset from ground edge.)
    - *fmin* = `1.5e9` (lowest calculation frequency.)
    - *fmax* = `4e9` (highest calculation frequency.)
    - *stub_length* = `18.4` (length of the stub.)
    - *stub_offset* = `41.4` (distance from the ground edge to stub.)
3. Create dielectric media:
    1. - **Label**: `air`
       - **Relative permittivity**: *1*
    2. - **Label**: `substrate`
       - **Relative permittivity**: *epsr*
4. Create the substrate layer.
    a) Create a cuboid.
    - **Base corner (C)**: `(0, 0, 0)`
    - **Width (W)**: *gnd_length*
    - **Depth (D)**: *gnd_length*
    - **Heigth (H)**: *substrate_height*
    - **Label**: *substrate*
5. Create the shielding box.
    a) Create a cuboid.
    - **Base corner (C)**: `(0, 0, 0)`
    - **Width (W)**: *gnd_length*

- **Depth (D)**: *gnd_length*
- **Heigth (H)**: *shielding_height*
- **Label**: `shielding_box`

6. Create the microstrip.

a) Create a cuboid.

- **Base corner (C)**: (*port_offset*, *strip_offset*, 0)
- **Width (W)**: *gnd_length-port_offset**2
- **Depth (D)**: *strip_width*
- **Heigth (H)**: *substrate_height*
- **Label**: *microstrip*

> **Note:** A cuboid is created whereby the bottom face will be coincident with the ground plane (bottom face of **shielding_box**). Alternately, a polygon or rectangle could have been used to create the microstrip.

7. Delete all the vertical faces of the **microstrip** part.

8. Create the stub.

a) Create a line.

- **Start point**: (*stub_offset*, *strip_offset+strip_width*, *substrate_height*)
- **End point**: (*stub_offset+strip_width*, *strip_offset+strip_width*, *substrate_height*)
- **Label**: *leading_edge*.

b) Sweep the line.

- **Start point**: (0, 0, 0)
- **End point**: (0, *stub_length*, 0)

c) Rename the sweep to `stub`.

9. Create the feed segments.

a) Create the first line.

- **Start point**: (0, *strip_offset+strip_width*/2, *substrate_height*)
- **End point**: (*port_offset*, *strip_offset+strip_width*/2, *substrate_height*)
- **Label**: *feed1*

b) Create the second line.

- **Start point**: (*gnd_length-port_offset*, *strip_offset+strip_width*/2, *substrate_height*)
- **End point**: (*gnd_length*, *strip_offset+strip_width*/2, *substrate_height*)
- **Label**: `feed2`

10. Union all the parts and rename the Union to `shielded_filter`.

11. Set the **Region** of the substrate to **substrate**.

12. Set the **Region** above the substrate to **air**.

13. Set the solution method on all regions to **FEM**.

> **Tip:** Open the **Modify Region** dialog and click the **Solution** tab. From the **Solution method** drop-down list, select **Finite Element Method (FEM)**.

14. Set all the face properties to PEC except for the two faces of the substrate (at the height of *substrate_height*).

15. Set the frequency
- **Continuous interpolated range**
- **Start frequency (Hz)**: *fmin*.
- **End frequency (Hz)**: *fmax*.

16. Add two FEM line ports, one for each feed line.

> **Tip:** From the **Source/Load** tab select **FEM Line Port**, use the option **Specify port as an edge** and click on the feed line in the 3D view.

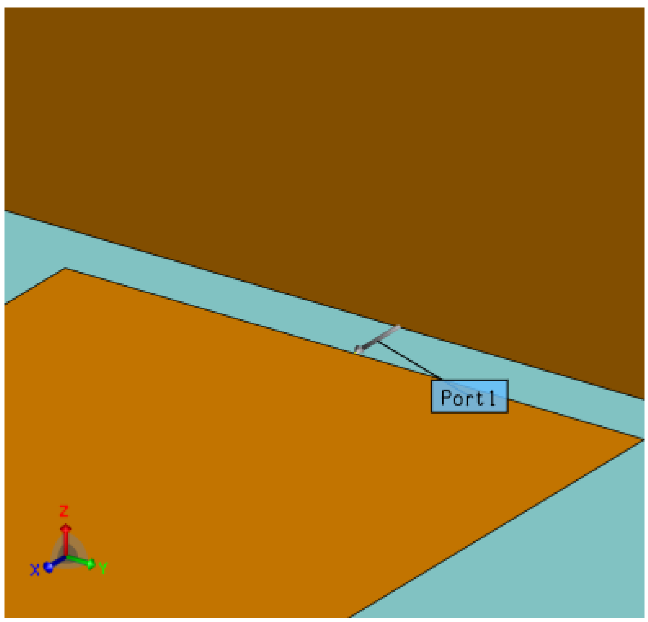


Figure 90: Zoomed in 3D view of one of the FEM line ports.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Create a **S-parameters** request.
   a) Set **FEMLinePort1** as the active port
   b) Add **FEMLinePort2** but set this port as inactive.

   > **Tip:** Every active port is treated as a new source increasing the runtime.

2. Delete **StandardConfiguration1**.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

E.1.2 Microstrip Filter on a Finite Substrate (SEP)

Model the microstrip filter using the surface equivalence principle. Use a voltage source on an edge port for the source.

Modify the finite element model from Microstrip Filter on a Finite Substrate (FEM) to use the surface equivalence principle. Remove the line ports.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Request a standard configuration and delete the existing S-Parameter configuration.
2. Delete the FEM line ports.
3. Expand **shielded_filter** and delete **Feed1** and **Feed2**.
4. Set the region properties of both regions back to the the default, namely **MoM/MLFMM with surface equivalence principle (SEP) - default**.
5. Set the **air** region back to **Free Space**.
6. Create two vertical plates between the microstrip feeding edges and the ground plane.
   a) Set the **Selection type** to **Edges** and select the leading edges of the microstrip line (at the input and output locations of the model).
   b) Create a copy of these edges.
   c) Select the copied edges (parts in the tree) and use the Sweep tool to sweep the edges in the negative Z direction a distance of *substrate_height*.
7. Create edges for the edge ports on the vertical plates.
   a) Split the vertical plates on a perpendicular plane at a height exactly halfway between the microstrip line and ground plane.
8. Union all the parts in the tree.
9. Create edge ports on the edges separating the two halves of the vertical plates.

   > **Note:** Edge ports must be surrounded on all sides by the same medium - these cannot be on the surface of a finite dielectric.

10. Set all the face properties to PEC except for the boundary surface between the dielectric layers.
11. Set a local mesh size on the microstrip faces of `strip_width/2`.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Create a **S-parameters** request.
   a) Set **EdgePort1** as the active port
   b) Add **EdgePort2** but set this port as inactive.
2. Delete **StandardConfiguration1**.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

E.1.3 Microstrip Filter on an Infinite Substrate (Planar Multilayer Green's Function)

Model the microstrip filter using an infinite substrate and planar multilayer Green's function. Use a voltage source on a microstrip port for the source.

Modify the finite element model from Microstrip Filter on a Finite Substrate (FEM) to use an infinite substrate. Remove the line ports.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Request a standard configuration.
2. Delete the FEM line ports.
3. Expand **shielded_filter** and delete **Feed1** and **Feed2**.
4. Set the region properties of the two regions back to the default (**MoM/MLFMM with surface equivalence principle (SEP) - default**).
5. Delete all horizontally orientated faces, except that of the top of the box, the microstrip line and stub.
6. Add a planar multilayer substrate (infinite plane) with a conducting layer at the bottom.
    a) Select **Plane / Ground.**
        - Click **Planar multilayer substrate**.
        - **Thickness** (Layer 1): *substrate_height*
        - **Medium** (Layer 1): *substrate*
        - **Ground plane** (Layer 1): **PEC**
        - **Z value at the top of layer 1**: *substrate_height*
7. Add microstrip ports to the edges of the feedline (see Figure 91 and Figure 92).

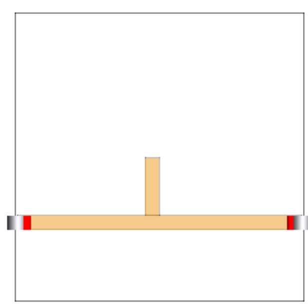


Figure 91: Microstrip ports were added to the edges of the feedline. Note that the infinite plane is hidden and a cutplane was added to show the port locations.

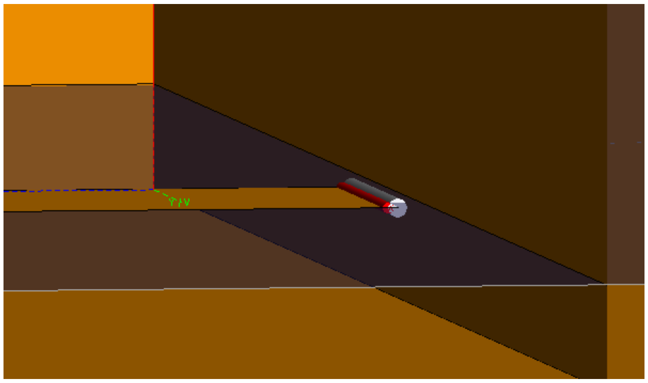


Figure 92: Zoomed in 3D view of one of the microstrip ports.

> **Tip:** The microstrip port connects to a single edge and is only used with infinite substrates. The positive terminal is indicated by the red cylinder in the 3D view.

8. Set a local mesh size on the microstrip lines (faces) of *strip_width*/2.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Create a **S-parameters** request.
    a) Set **MicrostripPort1** as the active port
    b) Add **MicrostripPort2** but set this port as inactive.
2. Delete **StandardConfiguration1**.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

E.1.4 Viewing the Results

View and post-process the results in POSTFEKO.

1. Plot the magnitude of S11 for all 3 models on the same **Cartesian** graph.
2. Create a **Duplicate view** of the graph and change the traces to display S21.

> **Note:** The different solution methods are in good agreement.

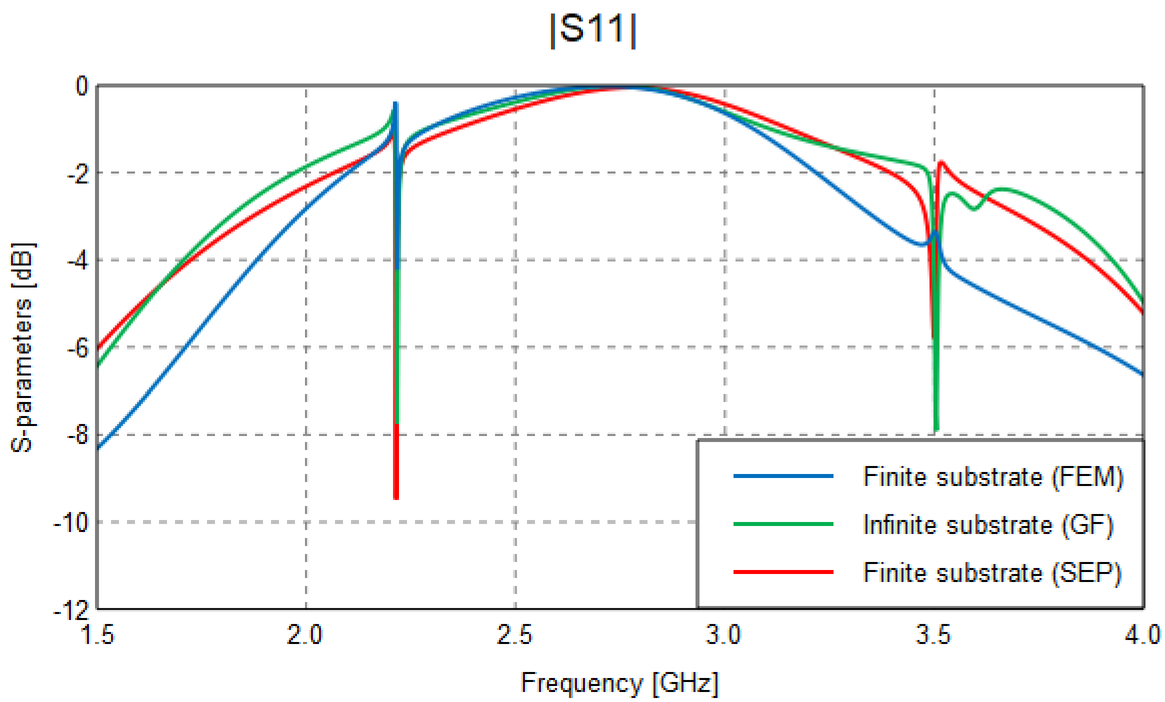


Figure 93: S11 in dB of the microstrip filter.

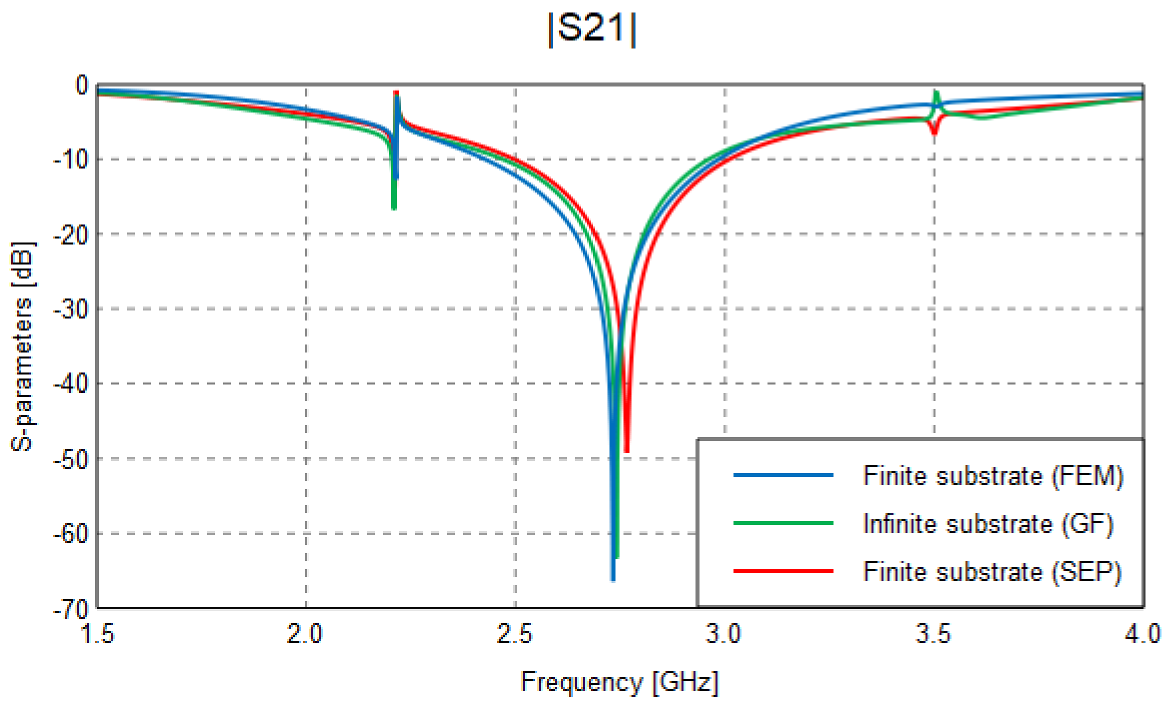


Figure 94: S21 in dB of the microstrip filter.

E.2 S-Parameter Coupling in a Stepped Waveguide Section

Calculate the transmission and reflection for a stepped waveguide transition from the Ku- to X-band. Use two solver methods, the method of moments (utilising waveguide ports) and the finite element method (utilising FEM modal ports).

The waveguide consists of a Ku-band section and an X-band section. Only the H_{10} mode with cutoff frequency, $f_c = \frac{c_0}{2a} = 9.4871$ GHz is considered.

The Ku-band section dimensions are as follows:

- a = 15.8 mm
- b = 7.9 mm

The X-band section dimensions are as follows:

- a = 22.9 mm
- b = 10.2 mm

Calculate S-parameters from the cutoff frequency to 15 GHz.

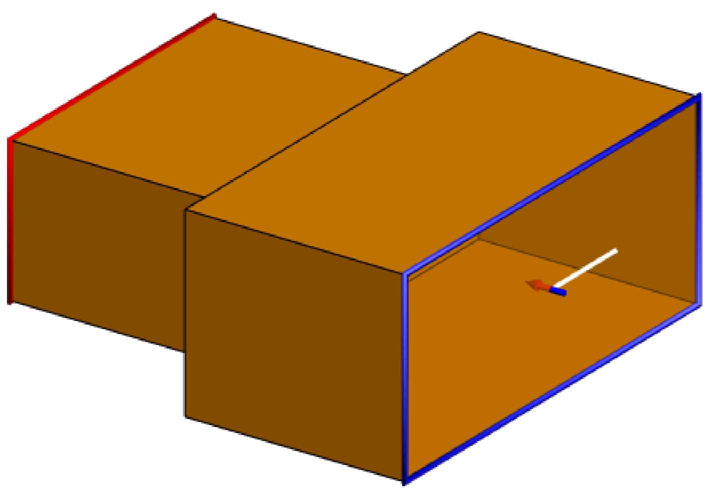


Figure 95: 3D view of the waveguide step.

E.2.1 S-Parameter Coupling in a Stepped Waveguide Section (MoM)

Calculate the transmission and reflection for the waveguide with the method of moments. Use waveguide ports for the sources.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to millimeters.
2. Define the following variables:
   - *a1* = `15.8` (width of Ku section.)
   - *b1* = `7.9` (height of Ku section.)
   - *l1* = `12` (length of Ku section.)
   - *a2* = `22.9` (width of X section.)
   - *b2* = `10.2` (height of X section.)
   - *l2* = `12` (length of X section.)
   - *fmin* = `9.4872e9` (minimum calculation frequency.)
   - *fmax* = `15e9` (maximum calculation frequency.)

   > **Note:** *fmin* is just above the cutoff frequency for the Ku band waveguide section.

3. Create the Ku band section.
   a) Create a cuboid
      - **Base corner (C)**: (-*a1/2*, -*l1*, -*b1/2*)
      - **Width (W)**: *a1*
      - **Depth (D)**: *l1*
      - **Height (H)**: *b1*
      - **Label**: `Ku_band_wguide`
4. Create the X band section.
   a) Create a cuboid.
      - **Base corner (C)**: (-*a2/2*, 0, -*b2/2*)
      - **Width (W)**: *a2*
      - **Depth (D)**: *l2*
      - **Height (H)**: *b2*
      - **Label**: `X_band_wguide`
5. Union both cuboids.
6. Set the both regions to free space.

> **Tip:** Free space changes the region from solid PEC to vacuum.

7. Simplify the part with the simplify tool.

> **Tip:** The simplify tool removes the face at the junction between the two cuboids. Alternately, use face selection to delete this face.

8. Rename the face at the end of the Ku band section to `Port1` and at the X band section to `Port2`.
9. Add waveguide ports to the faces with labels **Port1** and **Port2**.

> **Tip:** For accurate phase results set the reference vector with the same orientation for both ports. This is not required for the magnitude.

10. Check that the propagation direction of each port is set inwards.
11. Set the frequency.
    - **Continuous interpolated range.**
    - **Start frequency (Hz)**: *fmin*
    - **End frequency (Hz)**: *fmax*

> **Tip:** Set symmetry to save computational resources.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Create a **S-parameters** request.
    a) Set **WaveguidePort1** as the active port.
    b) Include **WaveguidePort2** but set this port as inactive.
    c) For the **Properties** tab choose **Fundamental**.

    > **Tip:** Every active port acts as a new source, increasing the runtime.

2. Delete the standard configuration.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Mesh size** equal to **Fine**.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

E.2.2 S-Parameter Coupling in a Stepped Waveguide Section (FEM)

Calculate the transmission and reflection through the waveguide with the finite element method. Use FEM modal ports for the source. Modify the model from S-Parameter Coupling in a Stepped Waveguide Section (MoM).

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Delete the waveguide ports.
2. Create a dielectric medium:
   - **Label**: `air`
   - **Relative permittivity**: *1*
3. Set the **Region medium** of the waveguide to **air**.

   > **Note:** The FEM solver uses tetrahedral volume meshing. To mesh the volume of the waveguide into tetrahedra, a dielectric of `air` is created.

4. Change the solver to use the FEM.

   > **Tip:** Open the **Modify Region** dialog and click the **Solution** tab. From the **Solution method** drop-down list, select **Finite Element Method (FEM)**.

5. Create FEM modal ports on the faces for **FEMModalPort1** and **FEMModalPort2**.
6. Set all the faces of the model, except the port faces, to PEC.

   > **Note:** FEM modal port faces must be of the dielectric type.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a **S-parameters** request.

a) Set **FEMModalPort1** as the active port

b) Add **FEMModalPort2** but set this port as inactive.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Mesh size** equal to **Fine**.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

E.2.3 Viewing the Results

View and post-process the results in POSTFEKO.

Plot S11 and S21 for both models on a Cartesian graph.

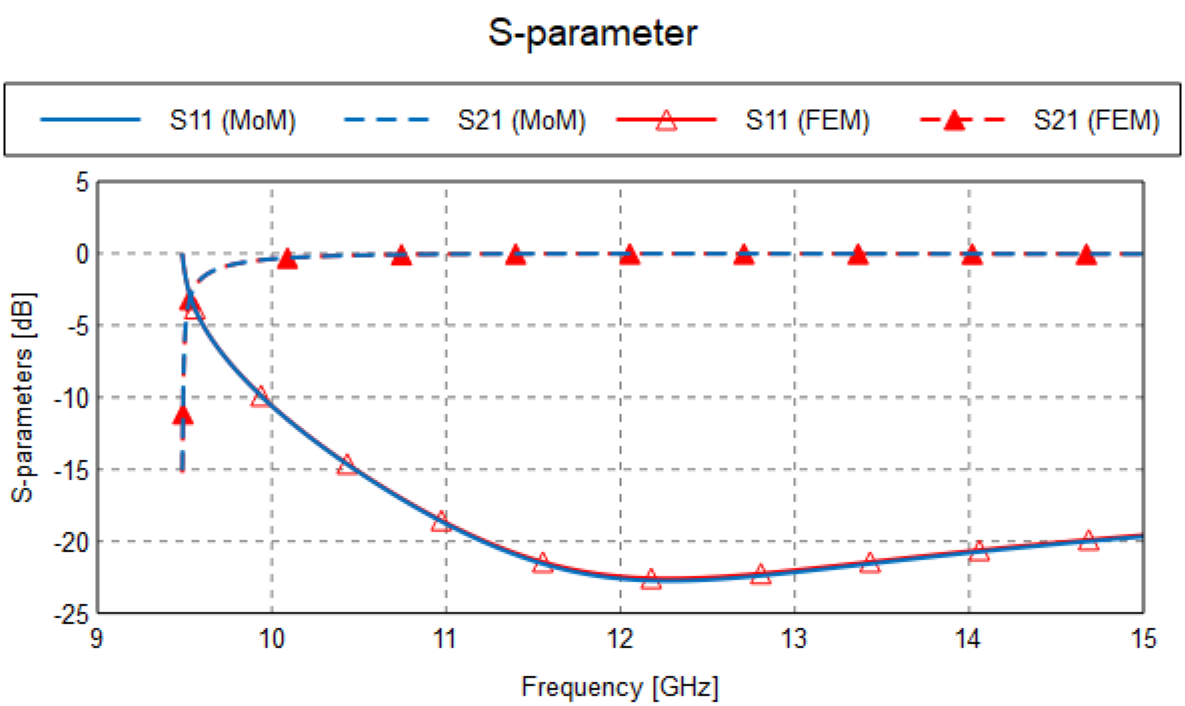


Figure 96: S-parameter results for the waveguide step in POSTFEKO.

The two solvers give identical results.

E.3 Using a Non-radiating Network to Match a Dipole Antenna

Match a short dipole for resonance at 1.4 GHz with an LC matching section. The matched network is modelled using a Spice circuit and S-parameters.

The dipole length is approximately 1/3λ. The matching network consists of a 2.43 pF shunt capacitor and 41.2 nH series inductor and is connected between the source and the dipole.

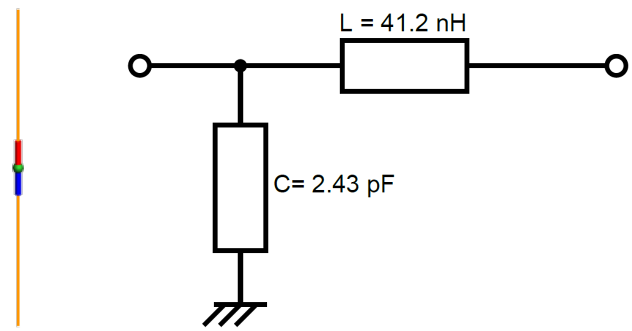


Figure 97: 3D view of the dipole (left) and the schematic representing the matching network used at the port (right).

> **Note:** The matching SPICE file (`Match_circuit.cir`) and Touchstone file (`Matching.s2p`) are included with the example.

E.3.1 Dipole Matching Using a SPICE Network

Match the dipole using a SPICE network.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to millimetres.
2. Define the following variables.
   - *fmin* = `1.3e9` (The lowest simulation frequency.)
   - *fmax* = `1.5e9` (The highest simulation frequency.)
   - *h* = `70` (The height of the dipole.)
   - *wireRadius* = `0.1` (The radius of the dipole segments.)
3. Create the dipole.
   a) Create a line.
      - **Start point**: (0, 0, -*h*/2)
      - **End point**: (0, 0, *h*/2)
      - **Label**: `dipole`

   b) Add a wire port to the middle of the line.

   c) Label the port `Port1`.
4. Set a continuous frequency range from *fmin* to *fmax*.
5. Create a general network using a SPICE circuit.
   a) Rename **GeneralNetwork1** to `MatchingNetwork`.

   > **Note:** The network label must correspond to the internal network name used in `Match_circuit.cir`.

```
Matching circuit
.SUBCKT MatchingNetwork n1 n2
c1 n1 0 2.43pF
l1 n1 n2 41.2nH
.ENDS NWN1
.end
```

6. Create a voltage source.
   a) For **Port** select **MatchingNetwork.Port1**
7. In the schematic view, connect Port1 (the dipole) to Port2 of the network.

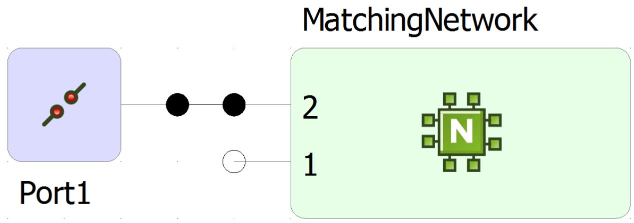


Figure 98: Schematic view of the matching network and port of the dipole.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

No solution requests are required.

> **Note:** Input impedance results are always available for voltage sources.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to *wireRadius*.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

E.3.2 Dipole Matching Using a General S-Parameter Network

Match the dipole using a general S-parameter network.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in Dipole Matching Using a SPICE Network and rename the file.
2. Modify the general network to define it in terms of S-parameters using a Touchstone file (`Matching.s2p`).

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for Dipole Matching Using a SPICE Network.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to *wireRadius*.

E.3.3 Viewing the Results

View and post-process the results in POSTFEKO.

Compare the reflection coefficient of the unmatched dipole and the matched dipole.

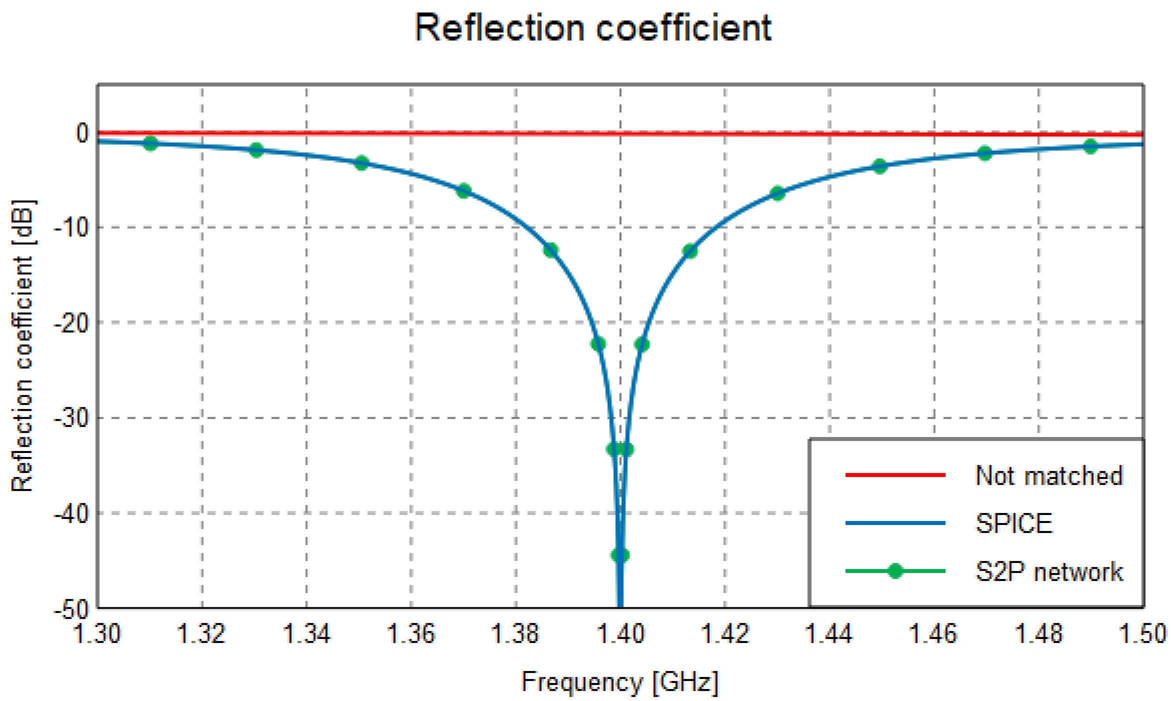


Figure 99: The reflection coefficient of the dipole before and after application of the feed matching.

Note: The reflection coefficient of the unmatched dipole is close to 0 dB over the frequency band. Hide the traces for the matched dipole to view the variation in the reflection coefficient of the unmatched dipole.

E.4 Subdividing a Model Using Non-Radiating Networks

Calculate the input impedance of a circularly polarised patch antenna fed through a microstrip branch coupler. Replace the branch coupler with a non-radiating network and compare with a full solution.

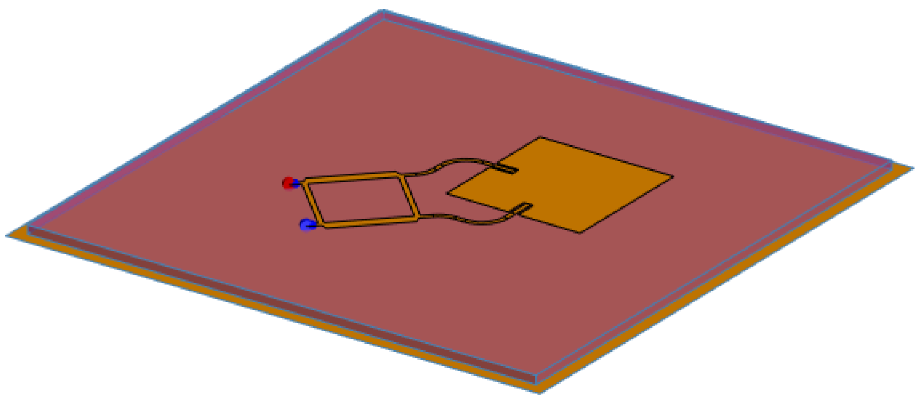


Figure 100: 3D view of the patch antenna with feed network.

Follow the steps below to solve the model:

1. Solve the S-parameters of the feed network separately and export to a Touchstone file.
2. Use the Touchstone file in a non-radiating network to feed the two input ports directly connected to the patch.
3. Solve the full model of the feed network and patch together.
4. Compare the results between the full model and subdivided model.

E.4.1 Feed Network

Calculate the S-parameters of the branch coupler and export to a Touchstone file. The branch coupler is designed for 120 Ω, distributes power evenly and has a 90 degree phase shift between the output ports.

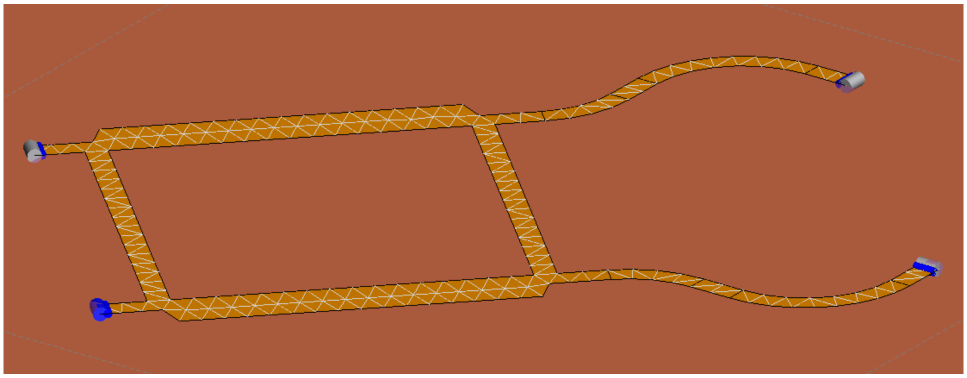


Figure 101: 3D view of the feed network.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Create a dielectric medium.
   - **Label**: `RogersDuroid5870`
   - **Relative permittivity**: *2.2*
   - **Dielectric loss tangent**: *0.0012*
2. Add a planar multilayer substrate (infinite plane) with a conducting layer at the bottom.
   a) Select **Plane / Ground.**
      - Click **Planar multilayer substrate**.
      - **Thickness** (Layer 1): *2.5e-3*
      - **Medium** (Layer 1): **RogersDuroid5870**
      - **Ground plane** (Layer 1): **PEC**
      - **Z value at the top of layer 1**: 0.
3. Create the branch coupler.
   a) Import the Parasolid model from file.

      > **Tip:** On the Home tab, select **Import** and select **Geometry**. Browse for the `feedNetwork.x_b` Parasolid file.
4. Create four microstrip ports on the four terminals of the network. Number the ports sequentially in an anti-clockwise manner.

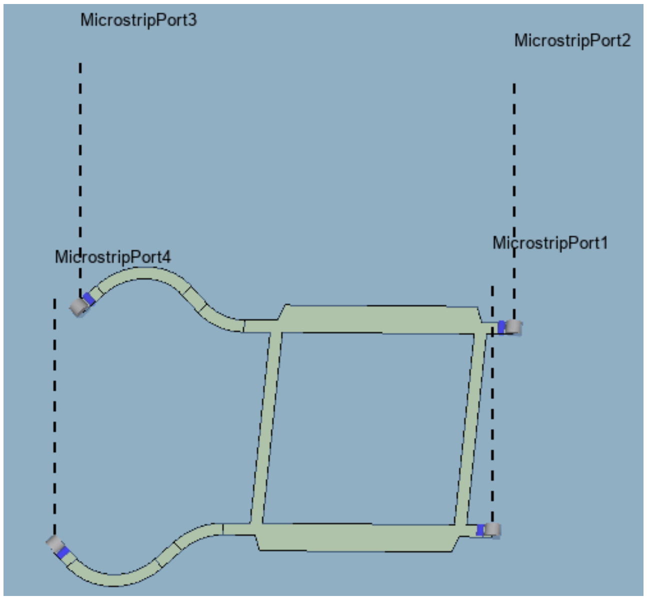


Figure 102: 3D view of the feed network showing the port numbering.

5. Add a 120 Ω load on **MicrostripPort4**.
6. Set the frequency.
   - **Continuous interpolated range**
   - **Start frequency (Hz)**: `0.8*2.4e9`
   - **End frequency (Hz)**: `1.2*2.4e9`
   - On the **Export** tab check the **Specify number of samples for exported data** check box and enter a value of `100`.

   > **Note:** The setting ensures that the exported Touchstone file contains 100 frequency samples.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create an **S-parameters** request.

a) Set **MicrostripPort1 to MicrostripPort3** as active ports.
b) For **Impedance** enter `120` for all the ports.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Mesh size** equal to **Custom**.

2. Set the **Triangle edge length** equal to `1.4e-3`.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

E.4.2 Patch with Non-Radiating Feed Network

Calculate the input impedance of the patch antenna with non-radiating feed network. Use the network parameters obtained from the Touchstone file.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Create a dielectric medium.
    - **Label**: `RogersDuroid5870`
    - **Relative permittivity**: *2.2*
    - **Dielectric loss tangent**: *0.0012*
2. Add a planar multilayer substrate (infinite plane) with a conducting layer at the bottom.
    a) Select **Plane / Ground.**
        - Click **Planar multilayer substrate**.
        - **Thickness** (Layer 1): *2.5e-3*
        - **Medium** (Layer 1): **RogersDuroid5870**
        - **Ground plane** (Layer 1): **PEC**
        - **Z value at the top of layer 1**: 0
3. Create the patch.
    a) Create a rectangle.
        - **Definition method**: **Base centre, width, depth**
        - **Base centre (C)**: `(0, 0, 0)`
        - **Width (W)**: `39e-3`
        - **Depth (D)**: `39e-3`
4. Create the inset feeds.
    a) Create a rectangle.
        - **Definition method**: **Base corner, width, depth**
        - **Base centre (C)**: `(-1.4e-3, -39e-3/2, 0)`
        - **Width (W)**: `2.8e-3`
        - **Depth (D)**: `6.5e-3`
        - **Label**: `InsetRectangleLarge`

    b) Create another rectangle.
        - **Definition method**: **Base corner, width, depth**
        - **Base centre (C)**: `(-1.4e-3/2, -39e-3/2, 0)`
        - **Width (W)**: `1.4e-3`
        - **Depth (D)**: `6.5e-3`
        - **Label**: `InsetRectangleSmall`

c) Union **InsetRectangleSmall** with **InsetRectangleLarge**.
d) Copy and rotate the Union by 90 degrees.
e) Union all the parts in the tree.
f) Delete the two face pairs from the newly created **Union1** to complete the inset feeds.

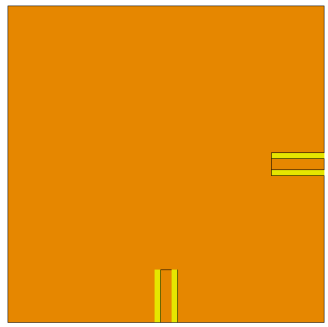


Figure 103: Construction of the inset feeds for the patch (redundant faces to be deleted in yellow.)

5. Create two microstrip ports on the outer edges of the inset feeds, one port for each feed.
6. Create a new network.
- **Data type**: **S-matrix**
- **Source**: **Touchstone file**
- **Number of network terminals**: `3`
- Browse for the `.s3p` file.

7. Connect the input ports of the patch to the output ports of the network in the schematic view.

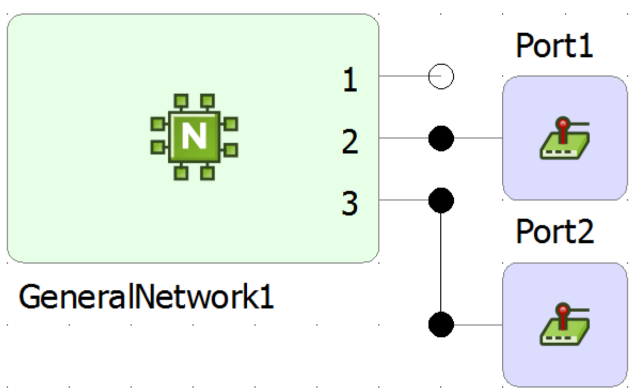


Figure 104: Schematic view showing the port connections.

8. Add a voltage source to **GeneralNetwork1.Port1**

> **Note:** The voltage source will not be displayed in the schematic view.

9. Set the frequency.
- **Continuous (interpolated) range**
- **Start frequency (Hz)**: `0.8*2.4e9`

- **End frequency (Hz)**: `1.2*2.4e9`

> **Note:** No output requests are necessary. The intput impedance of the voltage source is computed automatically.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Mesh size** equal to **Standard**.
2. Set a local mesh size on the two faces for the inset feeds of `1.4e-3`.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

E.4.3 Patch with Full Feed Network

Calculate the input impedance of the patch antenna and full feed network combined.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Open the model `touchstoneFedPatch.cfx`.
2. Save the model as `completePatch.cfx`.
3. Delete the voltage source.
4. Delete the general network connections in the schematic view.
5. Delete the general network.
6. Delete all the ports.
7. Import the model `feedNetwork.cfx`.
   a) On the **Import CADFEKO model** window select the following check boxes:
      - **Geometry**
      - **Meshing rules**
      - **Merge identical variables**
      - **Merge identical media**
8. Delete **MicrostripPort2** and **MicrostripPort3**.

   > **Note:** The output ports of the feed network are removed but the input ports are retained.

9. Union the feed network and patch.
10. Add a voltage source to **MicrostripPort1** (1 V, 0°, 50 Ω).
11. Add a 120 Ω load to **MicrostripPort4**.

   > **Note:** No output requests are necessary. The input impedance of the voltage source is computed automatically.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Mesh size** equal to **Standard**.
2. Set a local mesh size on all of the faces of the feed network and inset feeds of `1.4e-3`.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

E.4.4 Viewing the Results

View and post-process the results in POSTFEKO.

1. Load both the full model and subdivided model into POSTFEKO.
2. Plot the real and imaginary parts of the input impedance (MicrostripPort1) on a Cartesian graph.
3. Compare the computational resources of the models.

Table 4: Comparison of the computational resources for the models (per frequency point).

Model	Memory (MByte)	Time (s)
Full model	3.7	184
Network only	3.3	78
Patch with network	3.4	138

The solution time is reduced when substituting the feed network with a non-radiating network. This method reduces the design time of the full model if the feed network design is final but the antenna requires further changes. For larger models the computational resource differences will be more pronounced.

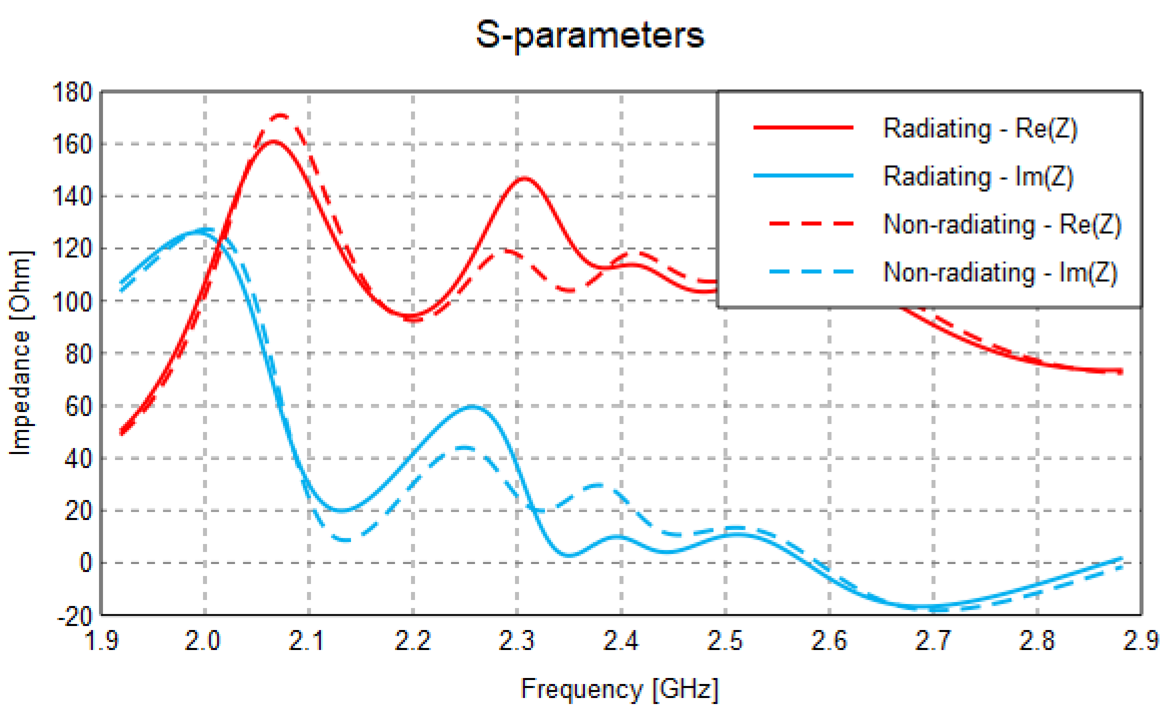


Figure 105: Input impedance of the patch: Full model vs using a non-radiating network.

E.5 Microstrip Coupler

Calculate the S-parameters (coupling) of a four port microstrip coupler. Use the finite difference time domain (FDTD).

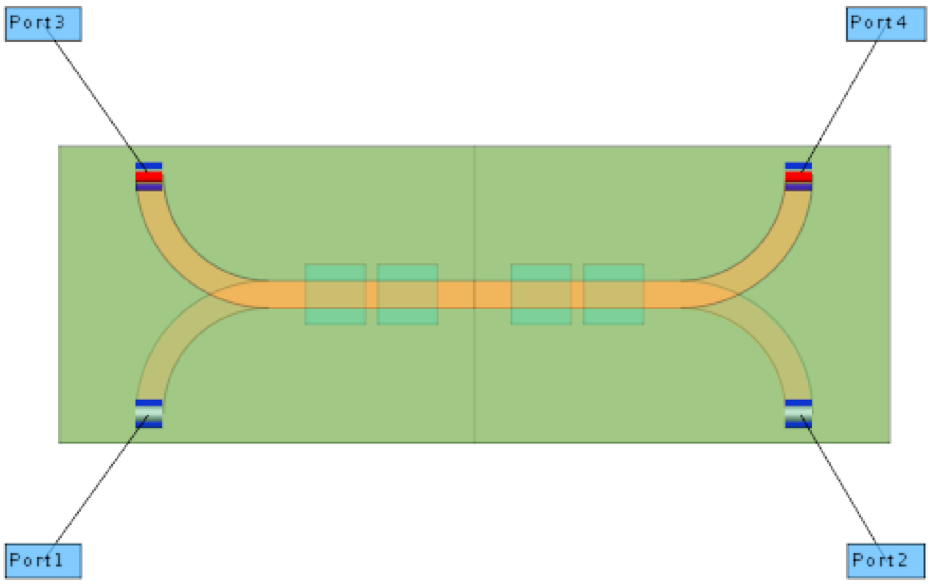


Figure 106: Transparent top view of the microstrip coupler containing multiple layers.

E.5.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to millimetres.
2. Define the following variables.
   - *d1* = 2.22 (Distance between apertures.)
   - *d2* = 12.51 (Distance between apertures.)
   - *epsr* = 2.2 (The relative permittivity of the substrate.)
   - *s* = 10 (Length of the aperture.)
   - *w* = 4.6 (Width of the microstrip.)
   - *strip_feed_arc_radius* = 2**s* (The radius of curved microstrip line.)
   - *strip_length* = 2* *s* + *d2* + *d1* (The straight section length of the microstrip line.)
   - *substrate_depth* = 50 (The substrate depth.)
   - *substrate_height* = 1.58 (The substrate height.)
   - *substrate_width* = 140 (The substrate width.)
   - *f_max* = 5e9 (The maximum frequency.)
   - *f_min* = 2.5e9 (The minimum frequency.)

3. Create a dielectric medium.
    - **Dielectric loss tangent**: `0`
    - **Relative permittivity**: *epsr*
    - **Label**: `substrate`
4. Create the straight section of the microstrip line.
    a) Create a rectangle.
        - **Definition method**: **Base corner, width, depth**
        - **Base corner (C)**: (0, -*w*/2, *substrate_height*)
        - **Width (W)**: *strip_length*
        - **Depth (D)**: *w*
5. Create the feed section of the microstrip.
    a) Create a rectangle.
        - **Definition method**: **Base corner, width, depth**
        - **Base corner (C)**: (0, 0, 0)
        - **Width (W)**: *w*
        - **Depth (D)**: *substrate_height*
        - On the **Workplane** tab set the **Origin**: (*strip_length*+*strip_feed_arc_radius*-*w*/2, *strip_feed_arc_radius*, 0)

    b) Rotate the workplane of the rectangle 90° around the U axis to align the rectangle in the XZ plane.

    > **Tip:** Right-click on the **Origin** box of the workplane and click **Rotate workplane**.
6. Create the arc section of the microstrip.
    a) Create an elliptic arc.
        - **Centre point (C)**: (*strip_length*, *strip_feed_arc_radius*, *substrate_height*)
        - **Radius (U)**: *strip_feed_arc_radius* + *w*/2
        - **Radius (V)**: *strip_feed_arc_radius* + *w*/2
        - **Start angle**: -90
        - **Stop angle**: 0
        - **Label**: `outer_circle`
7. Create a line.
    - **From**: (0, -*strip_feed_arc_radius*-*w*/2, 0)
    - **To**: (0, -*strip_feed_arc_radius*+*w*/2, 0)
    - On the **Workplane** tab set the **Origin**: (*strip_length*, *strip_feed_arc_radius*, *substrate_height*
8. Pathsweep **Line1** on **outer_circle**.
9. Union all parts.
    The resulting geometry represents half of the top microstrip section.

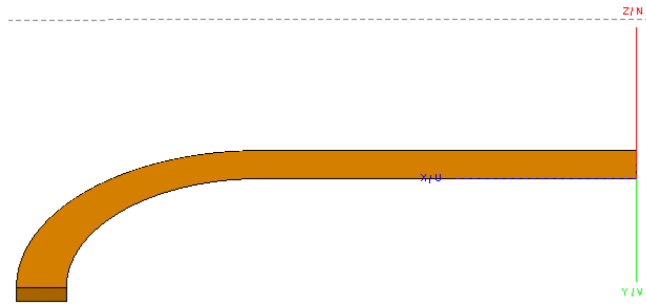


Figure 107: Geometry after Union operation.

10. Copy and rotate **Union1** by 180° around the U axis.

> **Note:** The new part represents half of the bottom microstrip.

11. Create the ground plate.

a) Create a rectangle.

- **Base Corner (C)**: (0, -*substrate_depth*/2, 0)
- **Width (W)**: *substrate_width*/2
- **Depth (D)**: *substrate_depth*
- **Label**: `ground_plate`

12. Create an aperture.

a) Create a rectangle.

- **Base corner (C)**: (*d2*/2, -*s*/2, 0)
- **Width (W)**:*s*
- **Depth (D)**:*s*
- **Label**: `aperture_1`

13. Create a second aperture.

a) Create a rectangle.

- **Base corner (C)**: (*d2*/2+*s*+*d1*, -*s*/2, 0)
- **Width (W)**:*s*
- **Depth (D)**:*s*
- **Label**: `aperture_2`

14. Subtract **aperture_1** and **aperture_2** from **ground_plate**.
The resulting geometry is a ground plane between two microstrip lines with two square holes.

15. Copy and mirror all geometry around the VN plane.

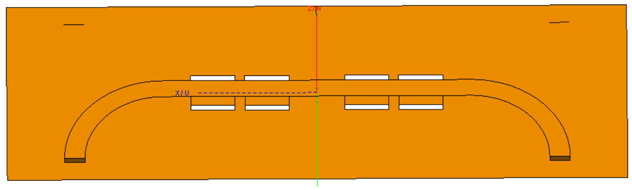


Figure 108: Geometry after the copy and mirror operation.

16. Union all the parts.

17. Set all faces to perfect electric conductor (PEC).

> **Tip:** Faces set to PEC remain PEC when becoming faces of a dielectric region.

18. Add edge ports.

Port1
: Define an edge port between the bottom microstrip feed on the negative X side and the ground plate.

Port2
: Define an edge port between the bottom microstrip feed on the positive X side and the ground plate.

Port3
: Define an edge port between the top microstrip feed on the negative X side and the ground plate.

Port4
: Define an edge port between the top microstrip feed on the positive X side and the ground plate.

19. Create two substrate layers.

a) Create a cuboid to construct the top layer.

- **Definition method**: **Base centre, width, depth, height**
- **Base centre (C)**: (0, 0, 0)
- **Width (W)**: *substrate_width*
- **Depth (D)**: *substrate_depth*
- **Height (H)**: *substrate_height*
- **Label**: *top_layer*

b) Create a second cuboid to construct the bottom layer.

- **Definition method**: **Base centre, width, depth, height**
- **Base centre (C)**: (0, 0, -*substrate_height*)
- **Width (W)**: *substrate_width*
- **Depth (D)**: *substrate_depth*
- **Height (H)**: *substrate_height*

- **Label**: *bottom_layer*

c) Union **top_layer** and **bottom_layer**.

d) Set both regions for this Union to the dielectric, **substrate**.

20. Union all the parts in the model.

21. Activate the FDTD solver.

> **Tip:** Open the **Solver settings** dialog and click the **FDTD** tab. Select the **Activate the finite difference time domain (FDTD) solver** check box.

22. Set the frequency.

- **Linearly spaced discrete points**
- **Start frequency (Hz)**: *f_min*
- **End frequency (Hz)**: *f_max*
- **Number of frequencies**: `101`

E.5.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Add an **S-Parameter Configuration**.

a) Include all four ports with a 50Ω reference impedance.

b) Set only *Port1* as **Active**.

E.5.3 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

E.5.4 Viewing the Results

View and post-process the results in POSTFEKO.

Plot the magnitude of S_{21}, S_{31} and S_{41} (in dB) on a Cartesian graph.

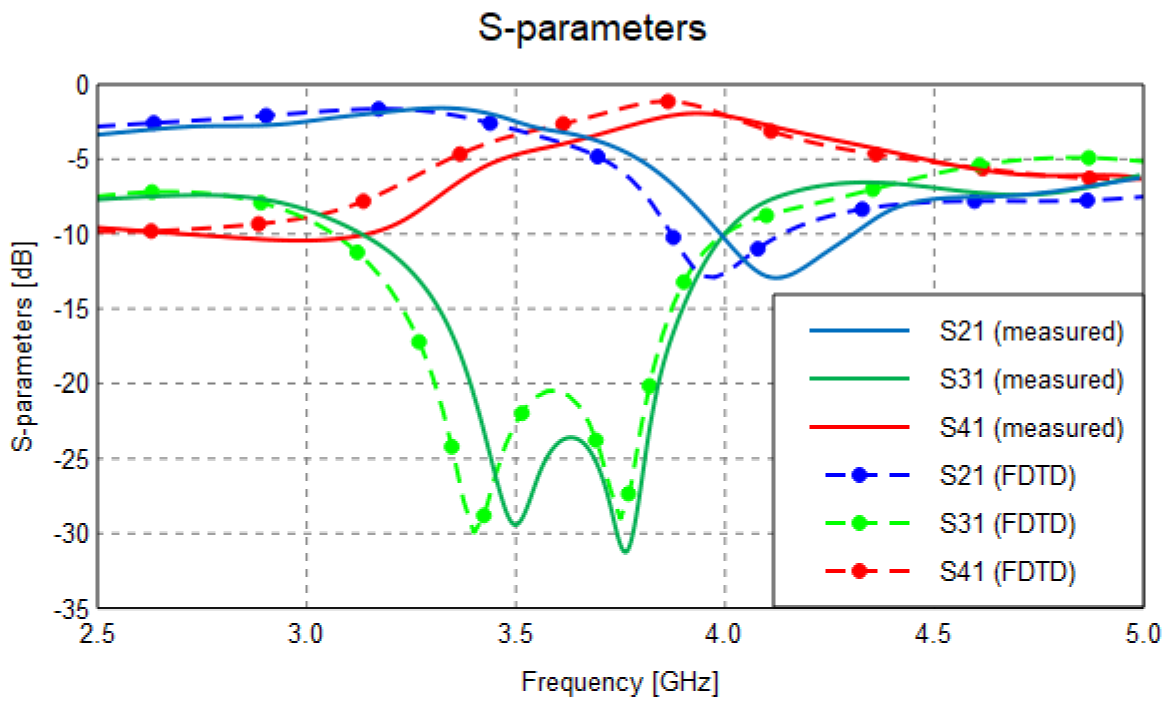


Figure 109: S_{21}, S_{31} and S_{41} of the microstrip coupler.

Compare the results with the literature reference, *On the design of planar microwave components using multilayer structures", by W. Schwab and W. Menzel, IEEE Trans. MTT, vol. 40, no. 1, Jan 1992, pp. 67-72, Fig 9.*

Note: The measured data from the referenced article are in good agreement with the simulated results. Any differences are most likely due to uncertainties regarding the model dimensions or dielectric properties.

Bio Electromagnetics

F

Simple examples demonstrating phantom and tissue exposure analsysis.

This chapter covers the following:

F.1 Exposure of Muscle Tissue Using the MoM/FEM Hybrid

Calculate the exposure for a sphere of muscle tissue illuminated by a dipole antenna.

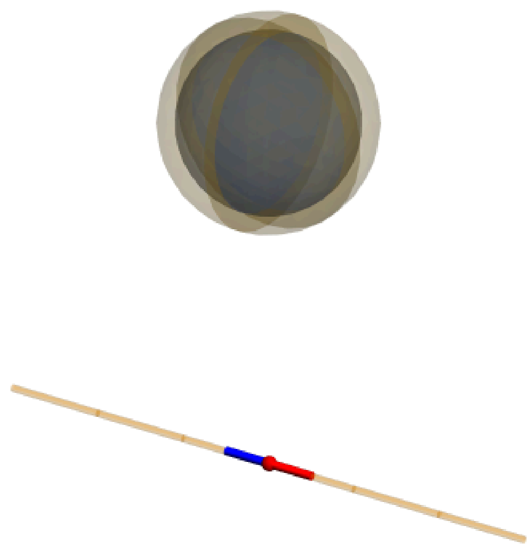


Figure 110: 3D view of the muscle tissue sphere, dipole antenna with a voltage source and single near field request point.

> **Note:** The model contains an air layer around the muscle sphere. The air layer is not required, but it reduces the number of triangle elements on the boundary between the FEM (dielectric sphere) region and MoM (free space region and dipole) region. Resource requirements are reduced when the number of boundary triangles are reduced. The computational resource benefit is strongly model dependent.

F.1.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables.
   - *freq* = 900e6 (The operating frequency.)
   - *f_min* = 100e6 (The minimum frequency.)
   - *f_max* = 1e9 (The maximum frequency.)
   - *d* = 0.1 (Distance between the dipole and muscle sphere.)
   - *rA* = 0.03 (Radius of the outer sphere.)

- *rM* = 0.025 (Radius of the inner sphere.)
- *lambda* = *c0*/*freq* (The wavelength in free space.)
- *wireRadius* = 1e-3 (Radius of the wire.)

2. Add the medium, **Muscle_Parallel_Fibers_Ovine** from the media library to the model.

3. Create a dielectric medium.
- **Dielectric loss tangent**: `0`
- **Relative permittivity**: *1*
- **Label**: `air`

4. Create the inner sphere.
- **Definition method**: **Centre, radius**
- **Centre**: (0, 0, 0)
- **Radius**: *rM*
- **Label**: `Muscle`

5. Create the outer sphere.
- **Definition method**: **Centre, radius**
- **Centre**: (0, 0, 0)
- **Radius**: *rA*
- **Label**: `Air`

6. Union the spheres, **Muscle** and **Air**.

7. Set the region of the inner sphere to **Muscle_Parallel_Fibers_Ovine**.

8. Set the region between the inner and outer sphere to **air**.

9. Set the solution method for the regions to FEM.

10. Create the dipole.

a) Create a line.
- **Start point**: (0, -*lambda*/4, -*d*)
- **End point**: (0, *lambda*/4, -*d*)

11. Add a wire port to the middle of the line.

12. Add a voltage source to the port. (1 V, 0°, 50 Ω).

13. Set the total source power (no mismatch) to 1 W.

14. Set a continuous frequency range from *f_min* to *f_max*.

F.1.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a near field request. The request is a single point located at the centre of the sphere.

- **Definition method**: **Cartesian**
- Select **Specify number of points** from the list.
- **Start**: (0, 0, 0)

- **End**: (0, 0, 0)
- **Number of field points**: (1, 1, 1)

F.1.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Wire segment radius** equal to *wireRadius*.

F.1.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

F.1.5 Viewing the Results

View and post-process the results in POSTFEKO.

View the electric field strength as a function of frequency.

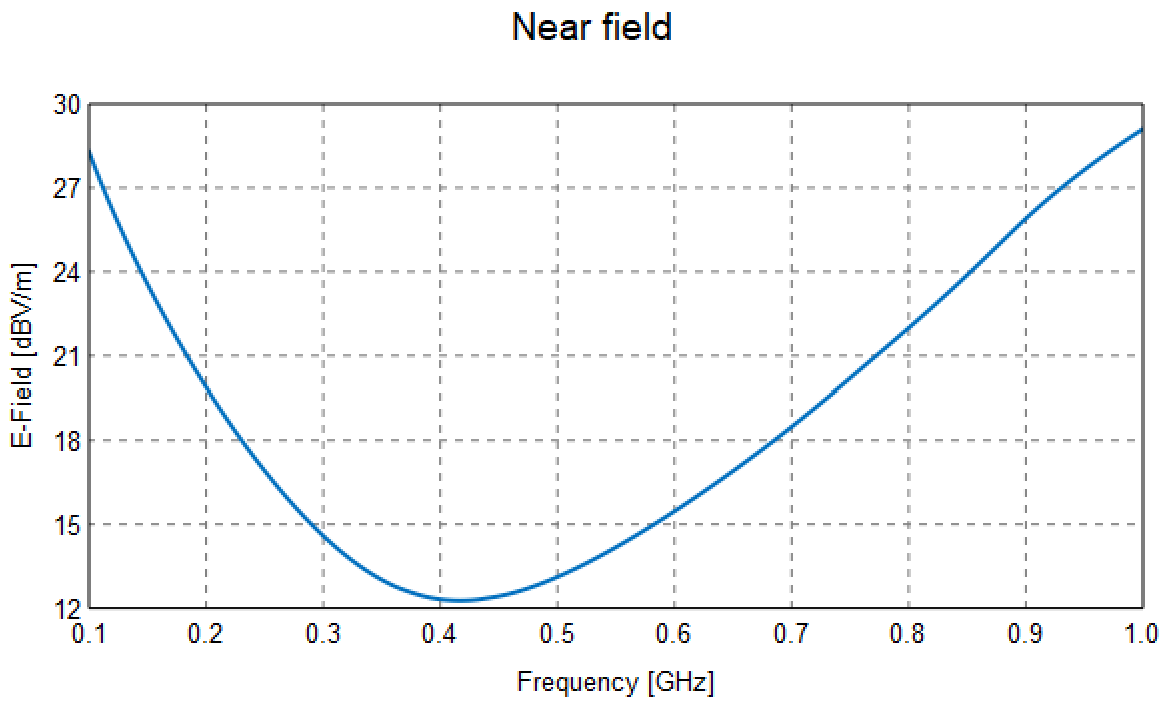


Figure 111: The electric field at the centre of the sphere over frequency.

F.2 Magnetic Resonance Imaging (MRI) Birdcage Head Coil Example

Calculate the fields, S-parameters and B quantities of an MRI birdcage head coil containing an elliptical phantom.

The coil is of the 7T highpass type with the tuning capacitors placed in the end-ring gaps between the 16 rungs.

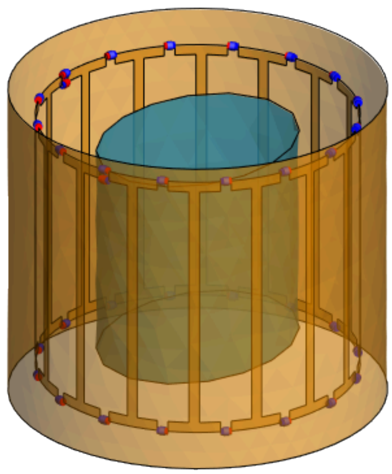


Figure 112: Geometry for the 7T head coil with elliptical phantom.

F.2.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

> **Note:** This model contains complex geometry. The creation steps are not provided, but the model is included in the Feko installation.

Note the following model details:

- The coil has an inner radius of 15 cm and an RF shield radius of 17.54 cm.
- The elliptical phantom has a major radius of 11 cm and a minor radius of 8.5 cm. Average head tissue properties are assigned to the phantom as follows:
  - relative permittivity: 36
  - conductivity: 0.657
- Capacitive loads (C = 4.15 pF) are added to the wire ports between the end-ring gaps on both sides of the birdcage.

Note: This example is solved with MoM (SEP), but MoM/FEM or FDTD could also be used.

F.2.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

The model uses multiple configurations.

- The first configuration is a standard configuration for calculating the currents.
- The second configuration is an S-parameter configuration.

1. Define the standard configuration (default configuration).
   a) Set the frequency for the standard configuration to 300 MHz.
   b) Add two voltage sources to the I and Q feed ports for the quadrature excitation.
      - The magnitude is set to 20 V for both ports.
      - A 90° phase delay is set on the Q port.
   c) Create a currents request (all currents).
   d) Create a near field request at Z=0.
2. Add an S-parameter configuration.
   a) Set a continuous frequency range for the S-parameter configuration from 290 MHz to 310 MHz.
   b) Set the ports **PortI1** and **PortQ1** to **Active.**

F.2.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use a custom mesh to accurately resolve the geometry.

1. Set the **Mesh size** equal to **Custom**.
2. Set the **Triangle edge length** equal to `4` cm.
3. Set the **Wire segment length** equal to *cap_length*.
4. Set the **Wire segment radius** equal to `0.01` cm.

F.2.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

F.2.5 Viewing the Results

View and post-process the results in POSTFEKO.

1. View the S-parameters on a Cartesian graph.

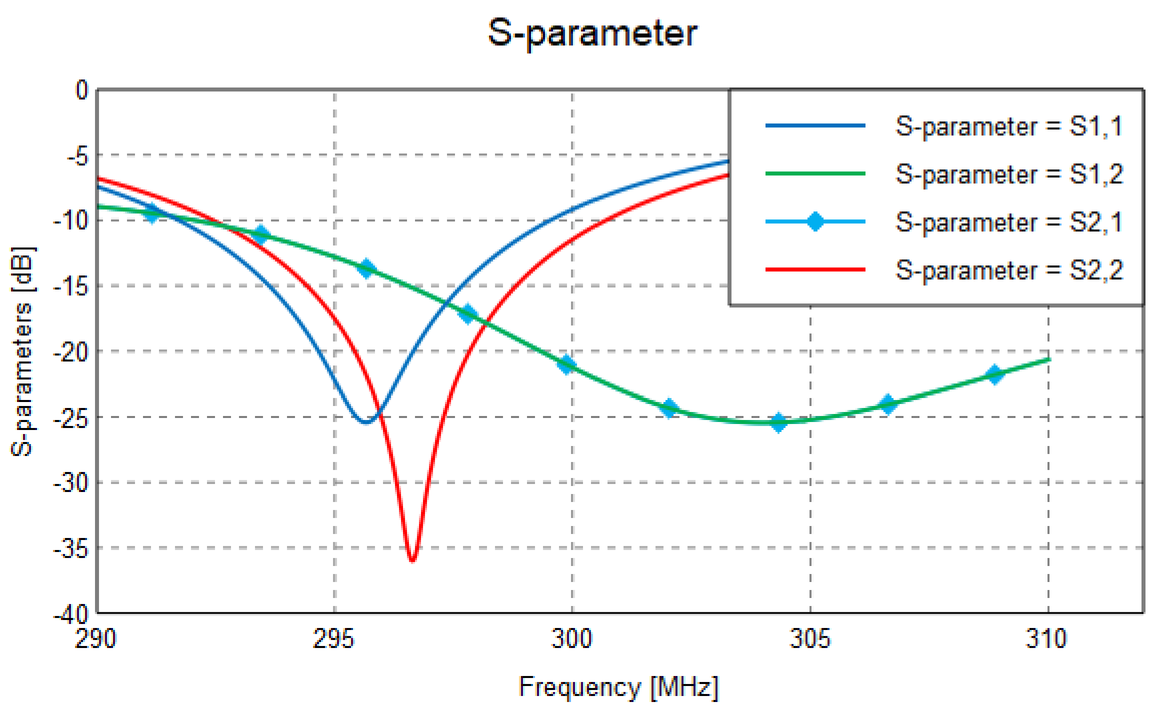


Figure 113: S-parameters for the I and Q feed ports of the coil.

2. View the currents in the 3D view and hide the phantom.

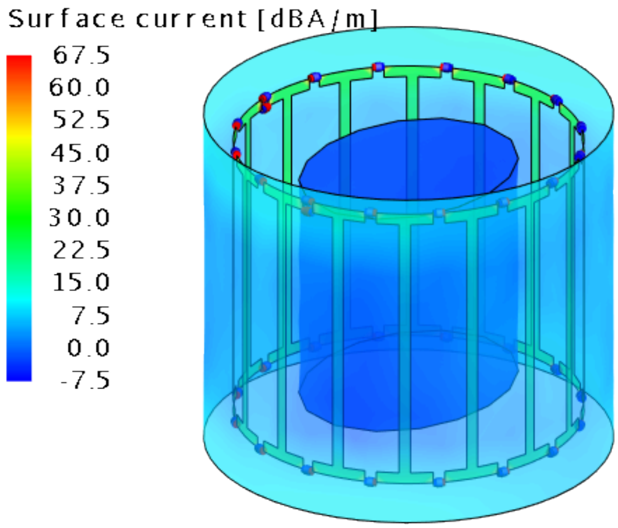


Figure 114: Surface currents of the coil rung with the phantom hidden.

The $B1^+$ and ratio ($B1^+/B1^-$) results are obtained from the MRI quantities automation script in POSTFEKO. The script is provided in the same folder as the CADFEKO model.

3. Add the results from the script to the 3D view.

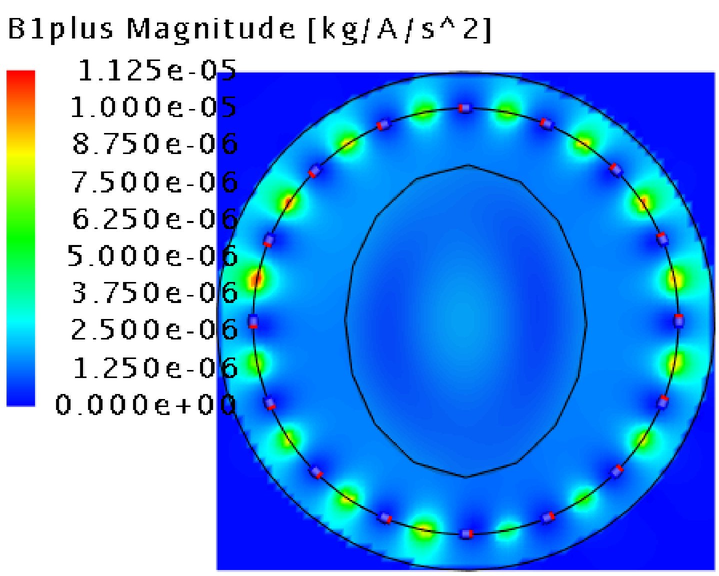


Figure 115: $B1^+$ field distribution at Z=0.

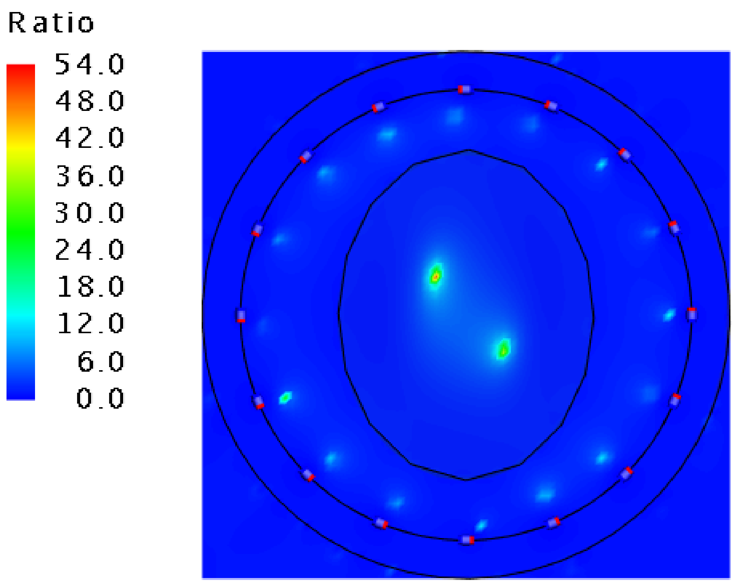


Figure 116: Ratio of ($B1^+/B1^-$)

Time Domain

G

A simple example demonstrating the time analysis of an incident plane wave on an obstacle.

This chapter covers the following:

G.1 Effect of Incident Plane Wave on an Obstacle Using Time Analysis

Observe the effect of an obstacle on a plane wave. Obtain frequency domain results using a wideband simulation using the method of moments (MoM). Perform post-processing of the frequency domain data to obtain a time response.

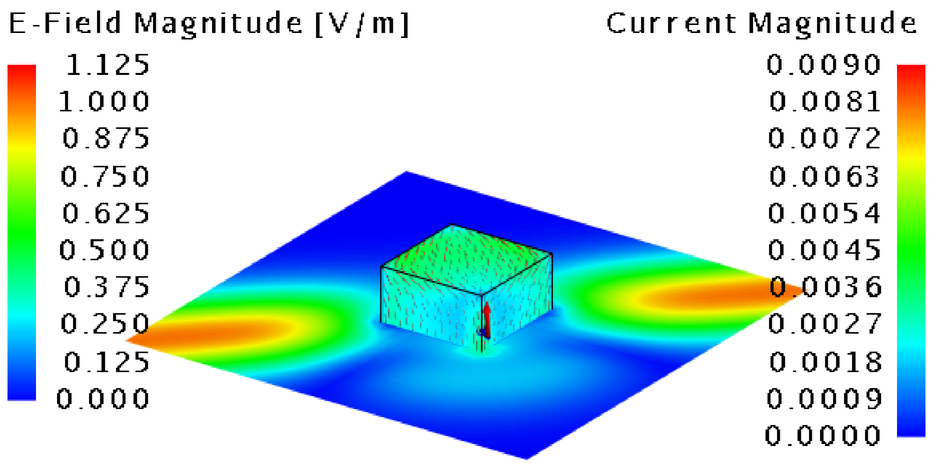


Figure 117: 3D view of the obstacle and time domain results.

> **Note:** A `.pfs` session file is included with the example. The time signals have been set up and you can view the near field results in the time domain.

G.1.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define a variable.
    - *d* = 1 (Length of the cuboid.)
2. Create a cuboid.
    - **Definition method**: **Base centre, width, depth, height**
    - **Base centre (C)**: (0, 0, -*d*/2)
    - **Width (W)**: *d*
    - **Depth (D)**: *d*
    - **Height (H)**: *d*

3. Add a single incident plane wave with θ=75° and ϕ=45°.
4. Set the frequency span between 2.5 MHz to 300 MHz using a list of discrete points.
   a) Import the list of discrete frequency points from the file `frequency_list.txt`.

G.1.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Create a currents request (all currents).
2. Create a near field request.
   - **Start**: (-2**d*, -2**d*, 0)
   - **End**: (2**d*, 2**d*, 0)
   - **Number of field points**: (31, 31, 1)
   - **Sample on edges**: enabled

G.1.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the mesh size equal to **Coarse**.

G.1.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

G.1.5 Viewing the Results

View and post-process the results in POSTFEKO.

The currents and near fields were calculated within a predetermined frequency range. Any time signal can be analysed provided its spectral content is within this range.

1. Create the input Gaussian pulse and triangular pulse using the following parameters:

Property	Gaussian pulse	Triangular pulse
Time axis unit	ns	ns
Total signal duration	100	100

Property	Gaussian pulse	Triangular pulse
Amplitude	1	1
Pulse delay	19	19
Pulse width	4	8
Number of samples	400	400

> **Tip:** On the **Time analysis** tab, in the **Time signal** group, click the **New time signal** icon.

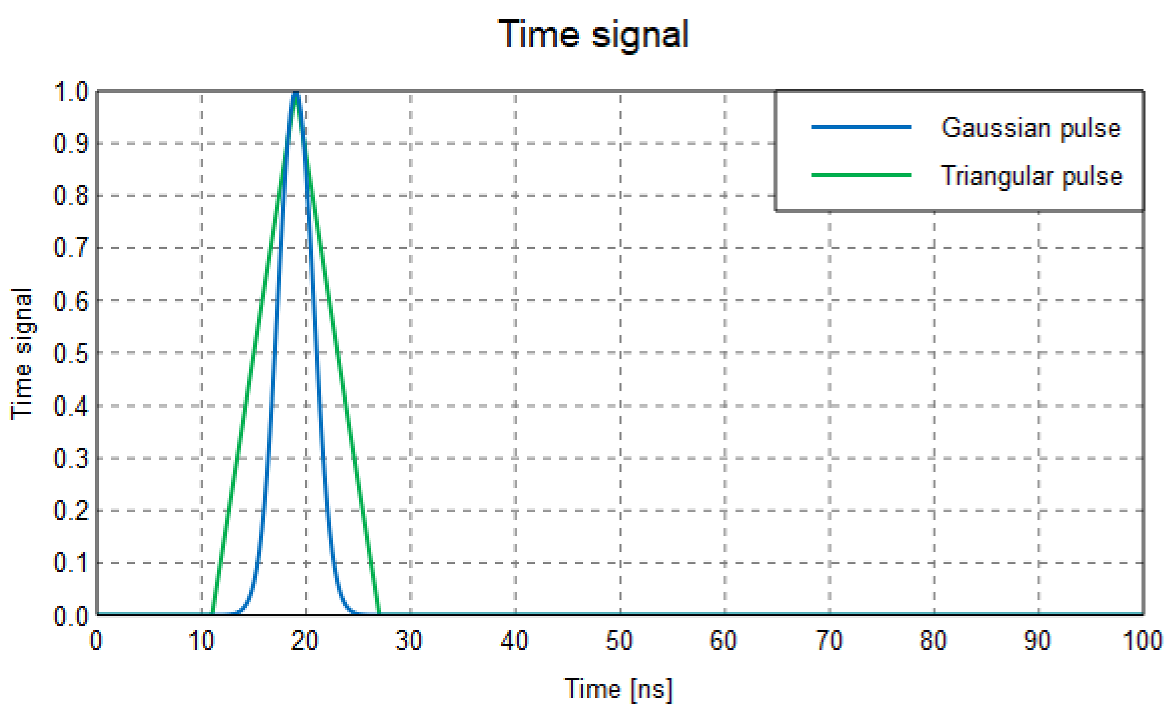


Figure 118: The Gaussian and triangular input signals.

2. View the time response of the system when a Gaussian pulse or a triangular pulse is applied.

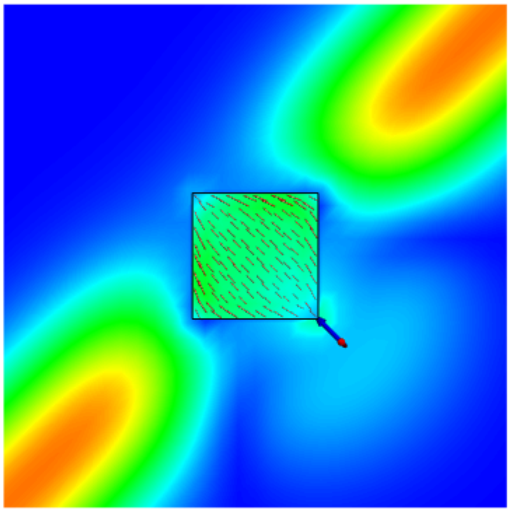


Figure 119: Near field time response for the Gaussian pulse after 19 ns.

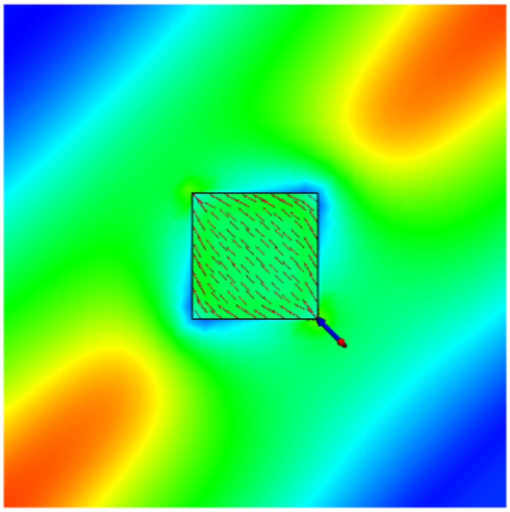


Figure 120: Near field time response for the triangular pulse (right) after 19 ns.

Tip: Gain insight into the time domain behaviour of a system using animation.

3. View the near field magnitude plotted over time at position (-2, -2, 0) m.

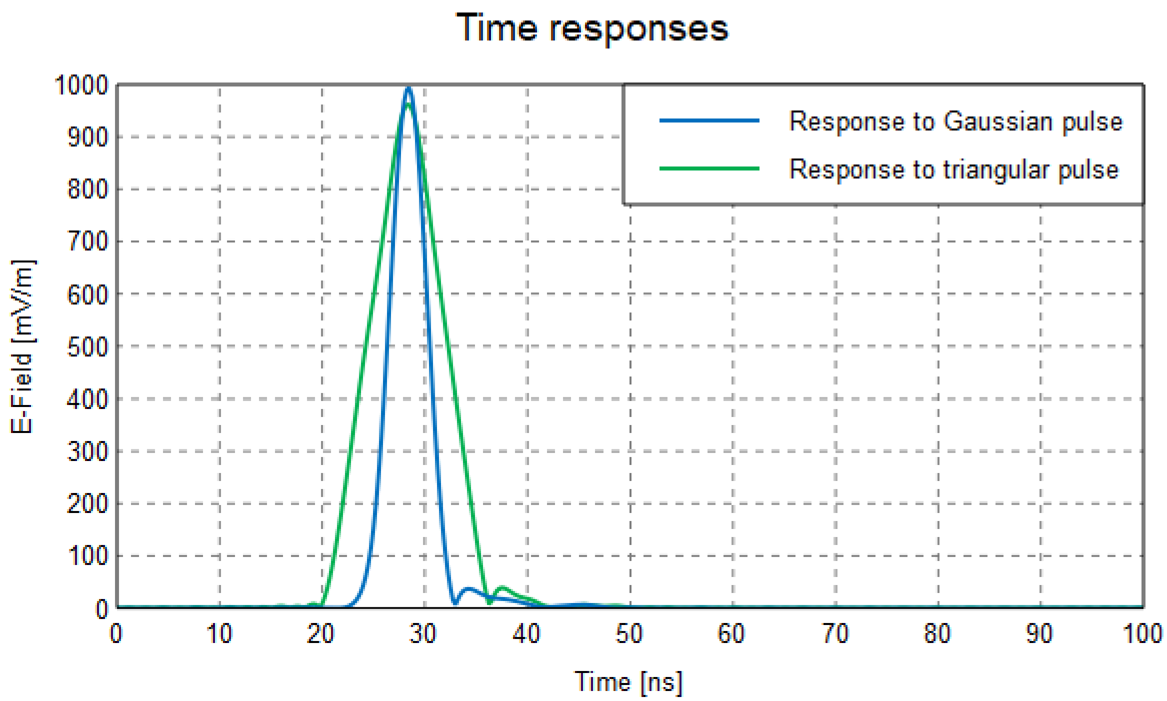


Figure 121: E-field magnitude at position (-2, -2, 0) m.

Special Solution Methods

H

Simple examples demonstrating using continuous frequency range, using the MLFMM for large models, using the LE-PO (large element physical optics) on subparts of the model and optimising the waveguide pin feed location.

This chapter covers the following:

H.1 Forked Dipole Antenna (Continuous Frequency Range)

Calculate the input admittance for a simple forked dipole.

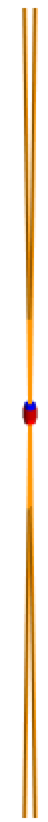


Figure 122: A 3D view of the forked dipole with a voltage source.

H.1.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables:
    - *fmin* = 100e6 (The minimum frequency.)
    - *fmax* = 300e6 (The maximum frequency.)
    - *wireRadius* = 1e-3 (Radius of the wire.)
2. Define the following named points.
    - *point1*: (-0.01, 0, 0.5)
    - *point2*: (0, 0, 0.01)
    - *point3*: (0.01, 0, 0.466)
    - *point4*: (0, 0, -0.01)

3. Create a line with the start and end coordinates of *point1* and *point2*.
4. Create a line with the start and end coordinates of *point2* and *point3*.
5. Copy and mirror the two lines around the UV plane.
6. Create a line with the start and end coordinates of *point2* and *point4*. Rename the label to `feed`.
7. Union all the lines.
8. Add a wire port to the middle of the line.
9. Add a voltage source to the port. (1 V, 0°, 50 Ω).
10. Set a continuous frequency range from *fmin* to *fmax*.

H.1.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

No solution requests are required.

> **Note:** Input impedance results are always available for voltage sources.

H.1.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Mesh size** equal to **Standard**.
2. Set the **Wire segment radius** equal to *wireRadius*.

H.1.4 Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

H.1.5 Viewing the Results

View and post-process the results in POSTFEKO.

View the input admittance (real and imaginary) of the voltage source on a Cartesian graph.

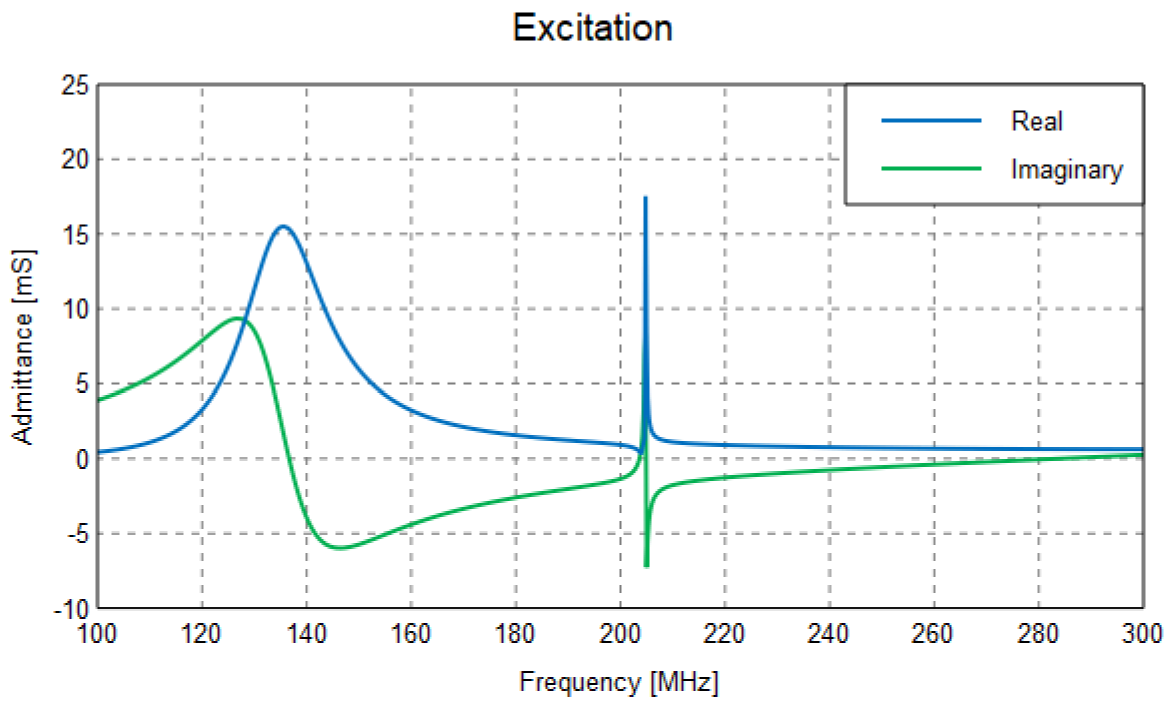


Figure 123: The input admittance (real and imaginary) of the forked dipole.

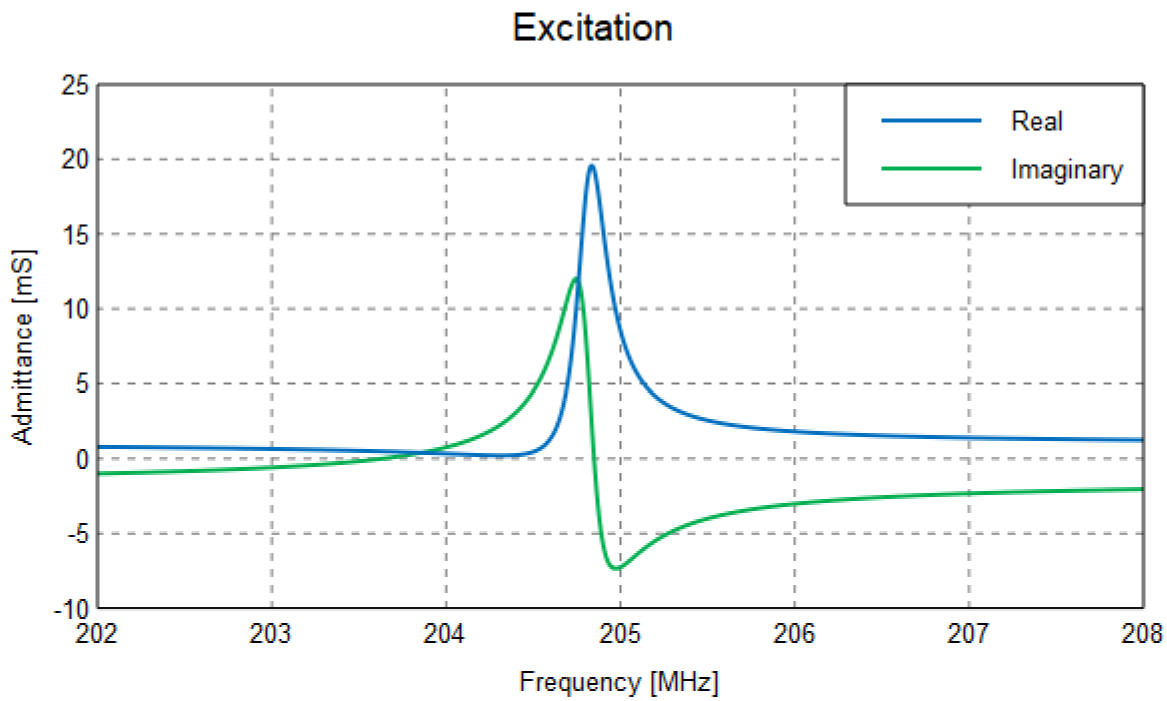


Figure 124: The input admittance (real and imaginary) of the forked dipole at the resonance point.

Compare the results with the literature reference, *Efficient wide–band evaluation of mobile communications antennas using [Z] or [Y] matrix interpolation with the method of moments", by K. L. Virga and Y. Rahmat-Samii, in the IEEE Transactions on Antennas and Propagation, vol. 47, pp. 65–76, January 1999.*

H.2 Using the MLFMM for Electrically Large Models

Consider the resource saving advantage of using the MLFMM for electrically large models

The size of the trihedral (13.5λ^2 surface area) is selected to allow the model to also be solved using the standard MoM.

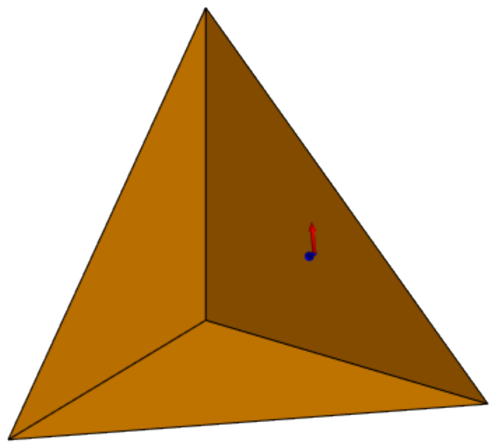


Figure 125: 3D view of the electrically large trihedral with an incident plane wave (source).

H.2.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables.
   - *lambda* = 1 (The wavelength in free space.)
   - *freq* = *c0/lambda* (The operating frequency.)
   - *s* = 3**lambda* (Side lengths of the trihedral.)

2. Create the trihedral.

   a) Create the first polygon.
      - **Corner 1**: (0, 0, 0)
      - **Corner 2**: (*s*, 0, 0)
      - **Corner 3**: (0, *s*, 0)

   b) Create the second polygon.
      - **Corner 1**: (0, 0, 0)
      - **Corner 2**: (0, 0, *s*)
      - **Corner 3**: (*s*, 0, 0)

   c) Create the third polygon.

- **Corner 1**: (0, 0, 0)
- **Corner 2**: (0, *s*, 0)
- **Corner 3**: (0, 0, *s*)

3. Union the plates.
4. Add a single incident plane wave with θ=60° and ϕ=45°.
5. Set the frequency to *freq*.
6. Solve the model with the MLFMM solver.

> **Tip:** Open the **Solver settings** dialog, click the **MLFMM / ACA** tab and then click **Solve model with the multilevel fast multipole method (MLFMM)**.

H.2.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a vertical far field request (-180°≤θ≤180°, with ϕ=45°). Sample the far field at θ=2° steps.

H.2.3 Viewing the Results

View and post-process the results in POSTFEKO.

1. View the runtime and memory using one of the following methods:
   - in the `*.out` file
   - in the model browser

   > **Tip:** In the model browser click **Results** tab and select **Solution Information**. View the solution information in the **details browser**.

2. Compare the runtime and memory for the MLFMM and MoM.

Table 5: Memory and runtime requirements for the MLFMM and MoM models.

Solution method	Memory (MBytes)	Runtime (seconds)
MoM	336	35
MLFMM	127	4

3. Compare the bistatic RCS of the trihedral for the MLFMM and MoM models.

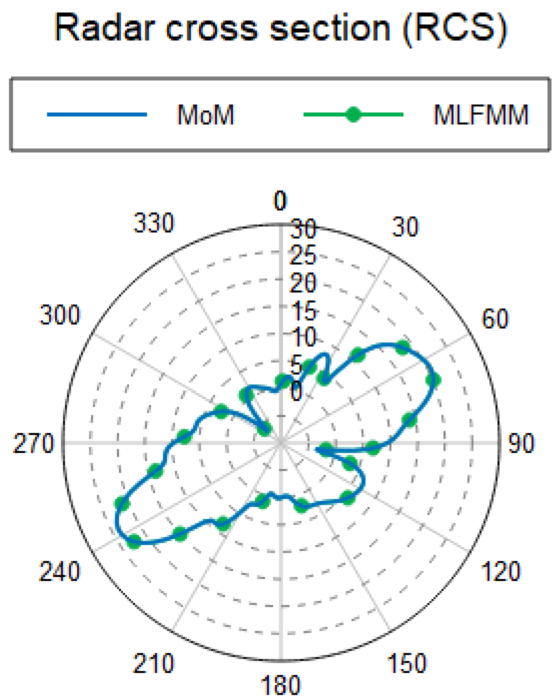


Figure 126: Comparison of the bistatic RCS for a trihedral using MLFMM and MoM .

H.3 Horn Feeding a Large Reflector

Calculate the gain for a cylindrical horn feeding a parabolic reflector at 12.5 GHz. The reflector is electrically large (diameter of 36 wavelengths) and well separated from the horn. Several techniques available in Feko are considered to reduce the required resources for electrically large models.

Use the following techniques to reduce the required resources:

- For electrically large models, use the multilevel fast multipole method (MLFMM) instead of the method of moments (MoM). The required memory is reduced considerably by using MLFMM.
- For subparts of the model, use large element physical optics (LE-PO) .
- Subdivide the problem and use an equivalent source.
    - Near field source - A region can be replaced by equivalent electric and magnetic field sources on the boundary of the region.
    - Spherical modes source - A far field can also be used as an impressed source.

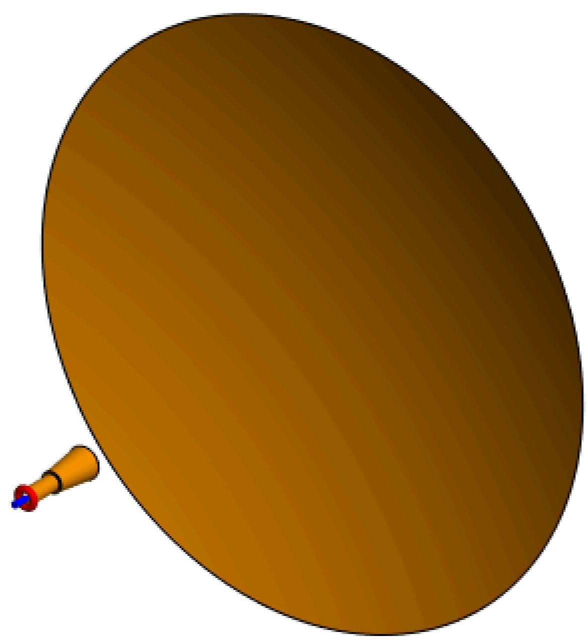


Figure 127: A 3D view of the cylindrical horn and parabolic reflector.

> **Tip:** Each model uses its predecessor as a starting point. Create the models in their presentation order. Save each model to a new location to keep them.

H.3.1 MoM Horn and LE-PO Reflector

Create the horn and the parabolic reflector. Solve the horn using MoM and the reflector using LE-PO.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Define the following variables.
   - *freq* = 12.5e9 (The operating frequency.)
   - *lam* = *c0/freq* (The wavelength in free space.)
   - *lam_w* = 0.0293 (The guide wavelength.)
   - *h_a* = 0.51**lam* (The waveguide radius.)
   - *h_b0* = 0.65**lam* (Flare base radius.)
   - *h_b* = *lam* (Flare top radius.)
   - *h_l* = 3.05**lam* (Flare length.)
   - *ph_centre* = -2.6821e-3 (Horn phase centre.)
   - *R* = 18**lam* (Reflector radius.)
   - *F* = 25*lam (Reflector focal length.)
   - *w_l* = 2**lam_w* (The waveguide length.)
2. Create the horn.
   a) Create a cylinder along the Z axis.
      1. **Definition method**: **Base centre, radius, height**
      2. **Base centre**: (0, 0, -*w_l*-*h_l*)
      3. **Radius (R)**: *h_a*
      4. **Height (H)**: *w_l*
      5. **Label**: `waveguide`

   b) Create a cone.
      - **Definition method**: **Base centre, base radius, top radius, height**
      - **Base centre**: (0, 0, -*h_l*)
      - **Base radius**: *h_b0*
      - **Height**: *h_l*
      - **Top radius**: *h_b*
      - **Label**: `flare`

   c) Union the two parts and simplify the resulting union.
   d) Rename the union of the two parts to `horn`.
   e) Delete the face at the front end of the horn to create an opening.
   f) Rotate the horn by -90°.
      - **Axis direction**: (0, 1, 0)

g) Create a waveguide port on the face at the back end of the waveguide section.

h) Add a waveguide source on the waveguide port. Use the default settings.

3. Create the parabolic reflector.

a) Create a paraboloid.

- **Base centre:** (0, 0, *F*)
- **Radius (R)**: *R*
- **Focal depth**: -*F*
- **Label**: `reflector`

a) Rotate the paraboloid by -90°.

- **Axis direction**: (0, 1, 0)

b) Set the solver method for the reflector face to use LEPO - always illuminated method.

> **Tip:** Open the **Modify Face** dialog and click the **Solution** tab. From the **Solve with special solution method**, select **Large element PO - always illuminated method**.

4. Decouple the MoM and LE-PO.

> **Tip:** Open the **Solver settings** dialog, click the **High frequency** tab and then click **Decouple PO and MoM solutions**.

> **Tip:** The decouple setting will save computational resources. It is not recommended where the coupling (interaction) between the PO and MoM objects is strong.

5. Set the frequency to *freq*.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a full 3D far field request (-180°≤θ≤180° and 0°≤ϕ≤180°). Sample the far field at θ=5° and ϕ=5° steps.

a) Enable the option, **Calculate continuous far field data**.

> **Tip:** Open the **Request/Modify far fields** dialog, click the **Advanced** tab and then select the **Calculate continuous far field data** check box.

> **Tip:** Spatially continuous far fields allow the far field to be re-sampled to any resolution in POSTFEKO.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Mesh size** equal to **Coarse**.

   > **Note:** Reduce simulation time for this example by using the **coarse** mesh setting. The standard mesh setting is recommended in general.

2. Set a local mesh size of *lam*/20 on the waveguide port face.

   > **Tip:** Open the **Modify Face** dialog, click the **Meshing** tab and then select the **Local mesh size** check box.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

H.3.2 MLFMM Horn and PO Reflector

Solve the horn using MLFMM and the reflector using PO.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in MoM Horn and LE-PO Reflector and rename the file.
2. Set the solver method for the reflector face to use PO - always illuminated method.

   > **Tip:** Open the **Modify Face** dialog and click the **Solution** tab. From the **Solve with special solution method** list, select **Physical optics (PO) - always illuminated**.

3. Solve the model with the MLFMM solver.

   > **Tip:** Open the **Solver settings** dialog, click the **MLFMM / ACA** tab and then click **Solve model with the multilevel fast multipole method (MLFMM)**.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for MoM Horn and LE-PO Reflector.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for MoM Horn and LE-PO Reflector.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

H.3.3 MLFMM Horn and LE-PO Reflector

Solve the horn using MLFMM and the reflector using LE-PO.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in MLFMM Horn and PO Reflector and rename the file.
2. Change the solver method for the reflector face to LEPO.

   > **Tip:** Open the **Modify Face** dialog and click the **Solution** tab. From the **Solve with special solution method**, select **Large element PO - always illuminated method**.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for MoM Horn and LE-PO Reflector.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for MoM Horn and LE-PO Reflector.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

H.3.4 Obtaining the Fields from the Horn Antenna

Calculate the fields of only the horn antenna. Export the fields to file for usage in an equivalent source.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Open the model from MoM Horn and LE-PO Reflector.
2. Save the model with a new name.
3. Delete the reflector.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

1. Modify the existing far field request to export spherical modes.
   a) On the **Advanced** tab, select the check box **Calculate spherical expansion mode coefficients**.
   b) Select the check box **Export spherical expansion coefficients to ASCII file**.
2. Create a near field request and export the fields to file.
   - **Definition methods**: **Spherical**
   - **Start**: (1.3**w_l*, 0, 0)
   - **End**: (1.3**w_l*, 180, 360)
   - **Increment**: (0, 5, 5)
   - Clear the **Sample on edges** check box.
   - On the Advanced tab, select the **Export fields to ASCII file** check box.

> **Tip:** Sampling on the edges would create duplicate request points at 0° and 360°.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for MoM Horn and LE-PO Reflector.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).

3. Address any errors and rerun the Solver (if applicable).

H.3.5 Near Field Source and LE-PO Reflector

Solve the horn using an equivalent near field source and the reflector using LE-PO.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in MoM Horn and LE-PO Reflector and rename the file.
2. Remove the **horn** part.
3. Create a near field data definition.
   - **E-field file**: Browse for the `.efe` file.
   - **H-field file**: Browse for the `.hfe` file.
   - **Coordinate system**: **Spherical**
   - **Radius (R)**: `1.3*`*w_l*
   - **Number of points along theta**: `36`
   - **Number of points along phi**: `72`
   - **Workplane origin**: (*w_l*, 0, 0)
4. Create a near field source that makes use of the near field data definition.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for MoM Horn and LE-PO Reflector.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for MoM Horn and LE-PO Reflector.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

H.3.6 Spherical Modes Source and LE-PO Reflector

Solve the horn using a spherical modes source and the reflector using LE-PO.

Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Use the model considered in MoM Horn and LE-PO Reflector and rename the file.
2. Remove the **horn** part.
3. Create the spherical modes field data definition.
    a) Browse for the `*.sph` file.
4. Create a spherical modes source that uses the spherical modes data definition.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Use the same calculation requests as for MoM Horn and LE-PO Reflector.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Use the same mesh settings as for MoM Horn and LE-PO Reflector.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

H.3.7 Comparing the Results

Compare the resource requirements and far field gain patterns for the models using different solver techniques.

1. Compare the resource requirements for the different techniques.

Table 6: Comparison of resources using different techniques. Generating the equivalent source data required negligible resources.

Model	RAM [MB]	Total Time [s]
MLFMM (reference)	4210	753
MLFMM Horn + PO Reflector	700	3663
MLFMM Horn + LE-PO Reflector	259	100
MoM Horn + LE-PO Reflector	540	161
Near field source + LE-PO Reflector	32	9
Spherical modes source + LE-PO Reflector	21	3

Note:
- Use LE-PO as the solution method for the reflector to reduce the memory requirement and solution time by several orders of magnitude.
- Subdivide the model into equivalent source models to reduce the resource requirements.

2. Compare the far field gain patterns for the models.

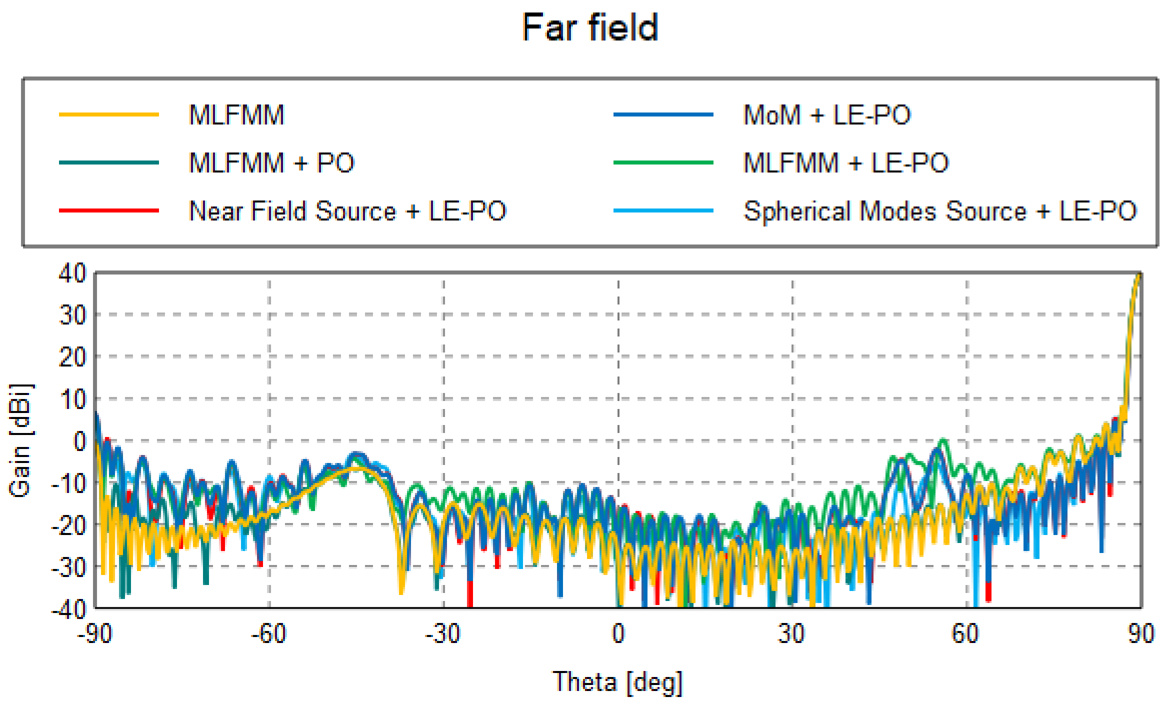


Figure 128: Gain of the reflector antenna calculated using different techniques over a 180 degree angle.

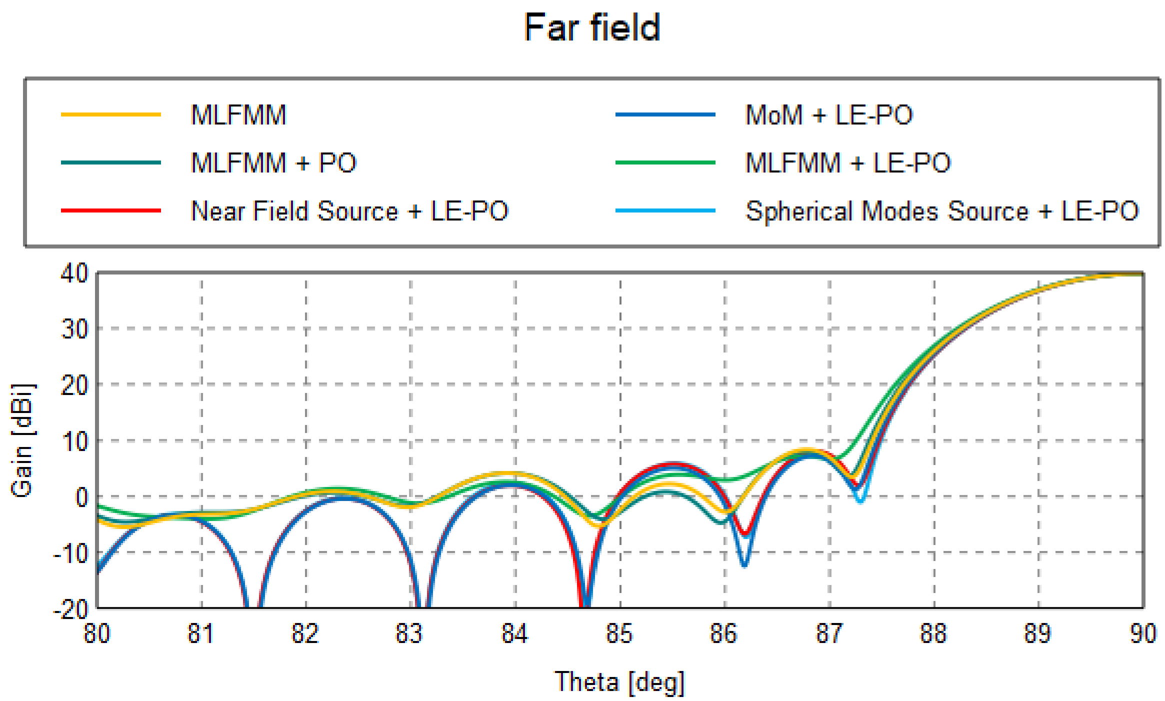


Figure 129: Gain of the reflector antenna calculated using different techniques - main lobe.

The difference in results is due to coupling between the horn and reflector that is only taken into account for the MLFMM solution. Although there is no restriction on the size of LE-PO triangles, the geometry must be accurately meshed.

For example, had a flat plate been used, only two triangles would have been required to obtain the same results.

H.4 Optimise Waveguide Pin Feed Location

Analyse the effect of the pin offset on the reflection coefficient using an optimisation grid search and the NGF solution settings.

A waveguide can be fed using a pin placed at a quarter of a waveguide wavelength from a terminated waveguide end. For an arbitrary cross-sectioned waveguide, the waveguide wavelength is not always known making it difficult to determine the pin feed position.

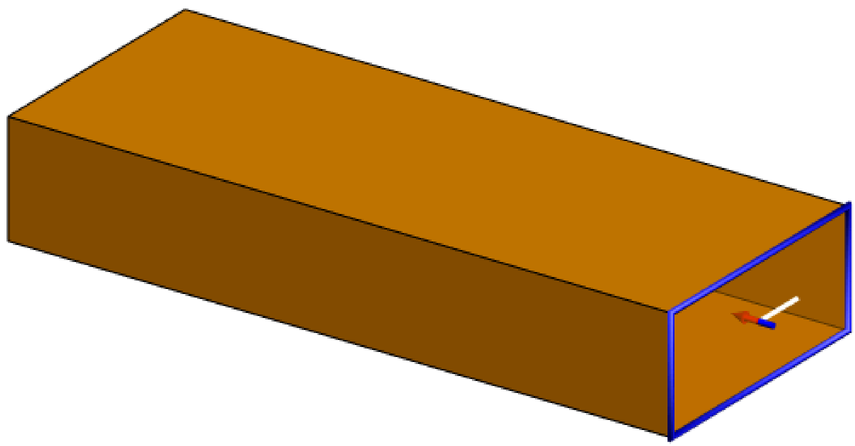


Figure 130: A 3D view of the waveguide fed with a pin feed and waveguide port.

H.4.1 Creating the Model

Create the model in CADFEKO. Define any ports and sources required for the model. Specify the operating frequency or frequency range for the model.

1. Set the model unit to millimetres.
2. Define the following variables:
   - *freq* = 10e9 (The operating frequency.)
   - *lambda* = *c0*/*freq**1e3 (The wavelength in free space. Note the unit is in millimetres.)
   - *n* = 1 (Feed pin position index.)
   - *pin_step_size* = *lambda*/32 (Distance between the pin positions.)
   - *pin_length* = 0.9**lambda*/4 (Length of pin feed monopole.)
   - *pin_offset* = *pin_step_size***n* (Pin offset from waveguide tip.)
   - *radius* = 0.1 (Radius of pin wires.)

- *waveguide_length* = *lambda**2 (Length of waveguide section.)
- *wr90_height* = 10.16 (Waveguide height for WR90 (X-Band, 8.2-12.4 GHz).)
- *wr90_width* = 22.86 (Waveguide width for WR90 (X-Band, 8.2-12.4 GHz).)

> **Note:** The model unit is millimetres. The variable *lambda* is not the actual wavelength but a parameter used in the geometry definition.

3. Create the waveguide.
   a) Create a cuboid.
      - **Definition method**: **Base centre, width, depth, height**
      - **Base centre**: (0, 0, 0)
      - **Width**: *wr90_width*
      - **Depth**: *waveguide_length*
      - **Height**: *wr90_height*
      - **Label**: `waveguide`
4. Set the **waveguide** region to **Free space**.
5. Imprint vertices on the mesh for the feed pin connections.
   a) Create a line.
      - **Start point**: (0, -*waveguide_length*/2, 0)
      - **End point**: (0, -*waveguide_length*/2 + *pin_step_size,* 0)
      - **Label**: `imprinted_edge_1`

   b) Copy and translate **imprinted_edge_1**.
      - **From**: (0, 0, 0)
      - **To**: (0, 2**pin_step_size*, 0)
      - **Number of copies**: 7
6. Union all geometry. Rename the resulting part to `waveguide_perforated`.
7. Activate the **NGF** and set *waveguide_perforated* as a static part.

> **Note:** The icon in the model tree, next to **waveguide_perforated**, indicates it is a static part and may not be edited.

8. Create the feed pin.
   a) Create a line.
      - **Start point**: (0, *pin_offset*, 0)
      - **End point**: (0, *pin_offset, pin_length*)
      - **Workplane origin**: (0, -*waveguide_length*/2, 0)
9. Add a wire port at the connection point between the feed pin and the waveguide floor.
10. Place a waveguide port at the opposite waveguide end. This will absorb the power injected by the pin feed.
    a) Rotate the reference direction with 90°.

11. Add a voltage source to the wire port. (1 V, 0°, 50 Ω).
12. Set the frequency to *freq*.

> **Note:** A magnetic plane of symmetry exists about the X=0 plane, but no computational performance benefit is obtained when used in conjunction with the NGF.

H.4.2 Defining Calculation Requests

Define the calculation requests in CADFEKO.

No solution requests are required. The input impedance and reflection coefficient are by default available as output for voltage sources in the model.

H.4.3 Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

1. Set the **Mesh size** equal to **Fine**.
2. Set the **Wire segment radius** equal to *radius*.

H.4.4 Adding an Optimisation Search

Add an optimisation search.

1. Add an optimisation search. Use the **Grid search** method and **Default number of points** equal to `15`.
2. Specify the optimisation parameters.
   a) Define the variable to be optimised.

Variable	Min value	Max value	Start value	Grid points
n	1	15	Empty	15

3. Define an **Impedance goal** to minimise the magnitude of the reflection coefficient for **VoltageSource1**.

H.4.5 Running the Optimisation

Run OPTFEKO to optimise the model according to requirements. During optimisation, OPTFEKO will call the Solver as required.

1. Run OPTFEKO.

2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun OPTFEKO (if applicable).

H.4.6 Viewing the Results

View and post-process the results in POSTFEKO.

For the first solver run, the full calculation needs to be performed. The static domain solution is stored to file and then re-used in subsequent solver iterations. By storing the solution to the static domain (for this example the **waveguide**), the Solver only needs to calculate the effect of the feed pin location in subsequent iterations.

1. View the resource requirements for each iteration.

Table 7: Comparison of resource requirements for each iteration.

Iteration	Memory (MBytes)	Runtime (seconds)
1	153	27
2	153	2
3	153	2
4	153	2
5	153	2
6	153	2
7	153	2
8	153	2
9	153	2
10	153	2
11	153	2
12	153	2
13	153	2
14	153	2
15	153	2

> **Note:** After the first iteration, the run time was significantly reduced. This effect of reduced run time after the first iteration becomes more pronounced as the size of the static domain increases relative to the dynamic domain.

2. View the reflection coefficient for each feed position on a Smith chart.

a) The port is optimally matched when the magnitude of the reflection coefficient is as small as possible or the input impedance is equal to 50 Ω. This condition is roughly met at iteration 6, see Figure 131. This corresponds to a feed position given by

$$pin\_offset \approx 6 \times pin\_step\_size = \frac{6}{32}\lambda \tag{14}$$

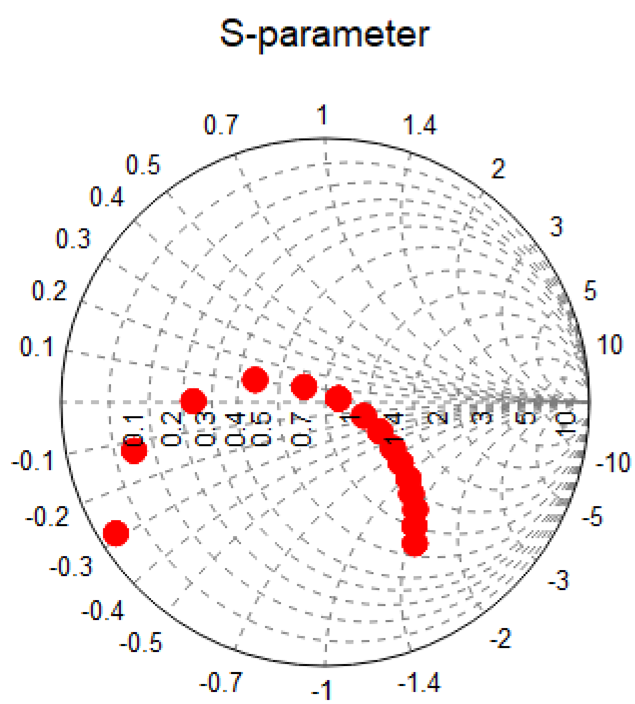


Figure 131: Smith chart showing the reflection coefficient for each feed pin position.

> **Tip:** Keep the temporary files. Then create the graph by plotting each model's source data.

H.5 Characterised Surfaces for FSS

Use characterised surfaces with ray launching geometrical optics (RL-GO) for an efficient solution of a frequency selective surface (FSS).

Characterised surfaces are defined with a `.tr` file that describes the transmission and reflection properties of the surface as a function of both frequency and angle of incidence (θ , ϕ). The first part of this example demonstrates how to create the `.tr` file for an FSS element. The second part of this example demonstrates how a characterised surface can be defined, using the `.tr` file.

> **Note:** Characterised surfaces are only supported with the RL-GO method.

H.5.1 Creating the FSS Model and Writing the .TR File

Generate a `.tr` file for an FSS element.

> **Note:** The **transmission / reflection** request is only supported with periodic boundary conditions or the planar Green's function.

Creating the Model

Set up the model to write the transmission / reflection coefficients `.tr` file to be used for the characterised surface.

1. Use the model created in Periodic Boundary Conditions for FSS Characterisation and rename the file.
2. Specify the symmetry about the X=0 and Y=0 planes as **Geometric symmetry**.
3. Set the frequency to 10 GHz.
4. Modify the plane wave source settings:
   - Click **Loop over multiple directions**.
   - Set the θ range from 0 to 180°.
   - Set the ϕ range from 0 to 360°.
   - Increment the incident angle, θ, in 6° steps.
   - Increment the incident angle, ϕ, in 6° steps.
   - Select the **Calculate orthogonal polarisations** check box.

   > **Tip:** Use a single theta cut for elements with small / no variation in ϕ.

   > **Tip:** For many θ and ϕ samples use the provided `.tr` file to save time.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Modify the **transmission / reflection coefficients** request.

a) Select the **Export transmission and reflection coefficients to file (*.tr)** check box to export the coefficients to file.

Modifying the Auto-Generated Mesh

Modify the model mesh in CADFEKO using the correct settings. A mesh is a discretised representation of a geometry model or mesh model used for simulation in the Solver.

Set the **Mesh size** equal to **Fine**.

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

Viewing the Results

View the results from the FSS element solution in POSTFEKO.

Plot the coefficients as a function of the plane wave theta and phi angles.

Tip: For (nearly) constant coefficients (such as in ϕ), use one θ cut to describe the surface.

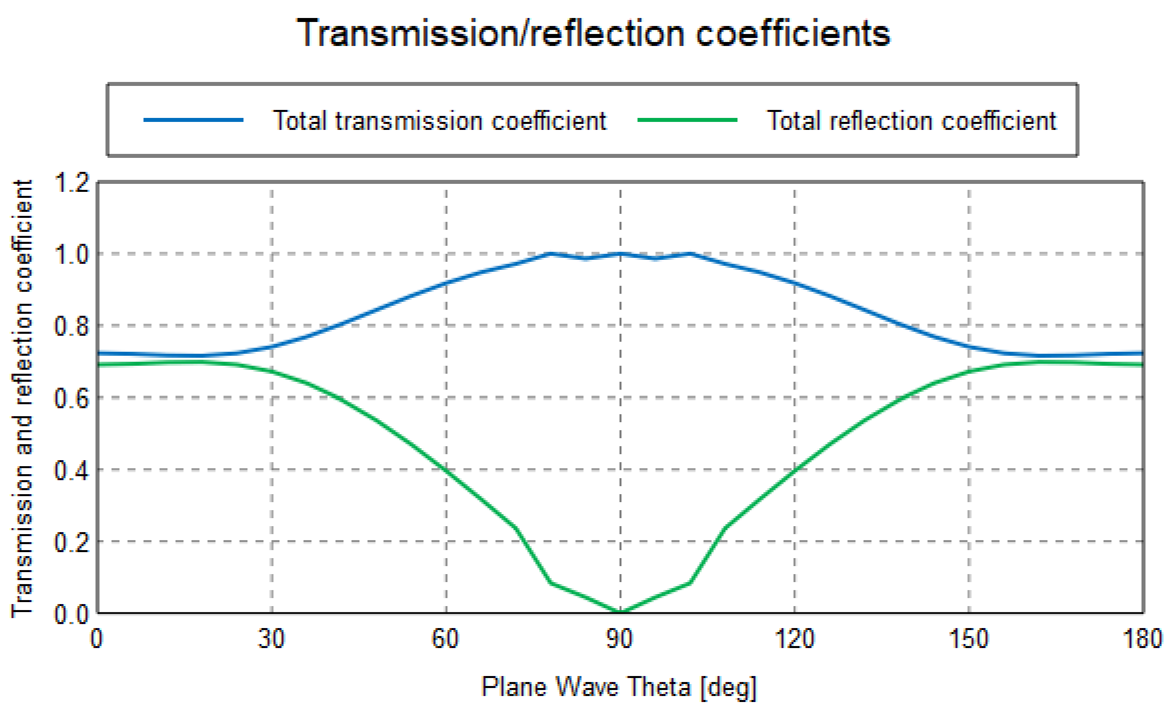


Figure 132: Transmission / reflection coefficient versus plane wave theta.

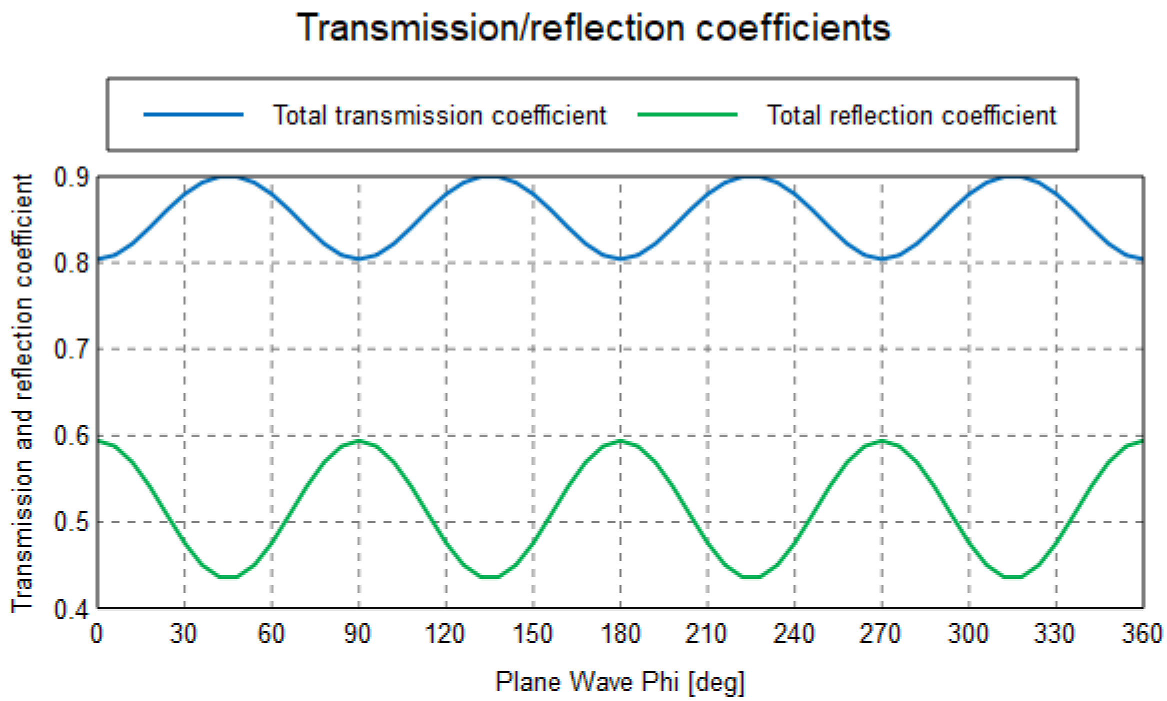


Figure 133: Transmission / reflection coefficient versus plane wave phi.

Note that the FSS elements have relatively good transmission through all θ and ϕ (> 0.7) at 10 GHz. The plots also show that sufficient resolution (6°) was used to capture the variations in θ and ϕ.

H.5.2 Creating the Characterised Surface

Use the `.tr` file and create a characterised surface for simulation with RL-GO.

Characterised Surface with RL-GO

Define a characterised surface that uses the `.tr` file.

Define a large rectangular plate where the characterised surface medium is set:

1. Create a new CADFEKO model.
2. Change the model unit to *mm*.
3. Define a new variable *d*=15.2 (this is the size of the FSS element used in Periodic Boundary Conditions for FSS Characterisation).
4. Define a rectangular plate.
   - **Base centre**: (0, 0, 0).
   - **Width** = *d**31.
   - **Length** = *d**31.
   - The plate size is equivalent to 31x31 FSS elements.
5. Create a **Characterised surface** medium using the `.tr` file that was created in Creating the FSS Model and Writing the .TR File.
6. Set the solver method for the rectangular plate to use RL-GO.

   > **Tip:** Open the **Modify Face** dialog and click the **Solution** tab. From the **Solve with special solution method** list, select **Ray launching - geometrical optics (RL-GO)**.

7. Apply the characterised surface to the rectangular plate.
   a) Medium: **CharacterisedSurface1**
   b) **U-vector**:
      1. **Start point**: (0, 0, 0)
      2. **End point**: (1, 0, 0)

   > **Tip:** Open the **Modify Face** dialog and click the **Solution** tab.

   > **Note:** Defining the U-Vector determines the orientation of the characterised surface elements on the face. In this example, the original orientation used for the FSS simulation is unchanged so the U-Vector is defined as X directed.

8. Set the **Solution frequency** to 10 GHz.
9. Define a plane wave source that loops over multiple directions with θ from 0 to 180° with and increment of 6°.

Defining Calculation Requests

Define the calculation requests in CADFEKO.

Create a near field request.

a) **Start**: (0, 0, 0)
b) **End**: (0, 0, *d**5)
c) **Number of field points**: (1, 1, 101)

Running the Feko Solver

Run the Solver to compute the calculation requests.

1. Run the **Solver**.
2. Investigate any warnings/errors found by the Solver (if applicable).
3. Address any errors and rerun the Solver (if applicable).

Viewing the Results

View and post-process the results from the characterised surface in POSTFEKO.

View the E-field above the characterised surface on a Cartesian graph. The results are compared to those calculated with a full solution using MLFMM where the geometry of the FSS element was duplicated to create the finite 31x31 element sheet.

Table 8: Computational requirements

31x31 FSS element sheet	Memory	Run time
Characterised surface - RL-GO	91 MB	9 sec
Full solution - MLFMM	332 MB	408 sec

Note the significant reduction in computational requirements over the MLFMM solution.

A practical example where the characterised surface is extremely useful is shown below: an FSS nosecone radome is represented using this feature. Due to the size and complexity of the structure, it becomes prohibitive to model and solve with a full wave solution. However it can be solved readily with the characterised surface approach.

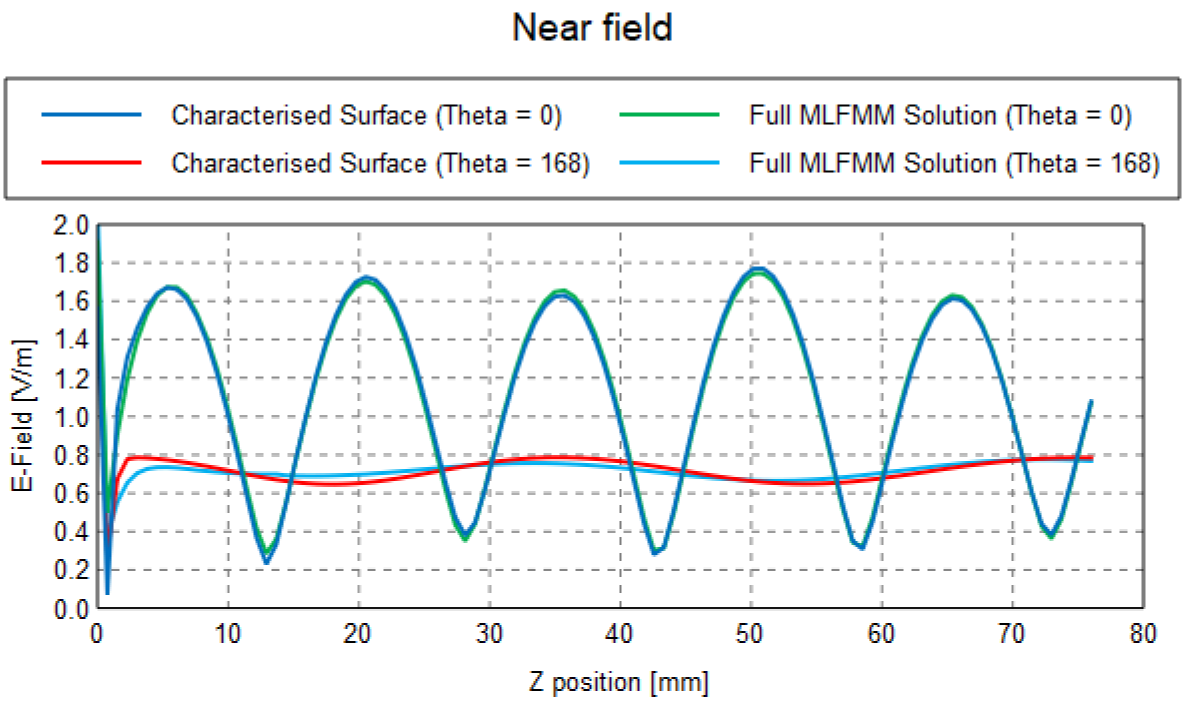


Figure 134: E-field above the characterised surface for 2 theta angles.

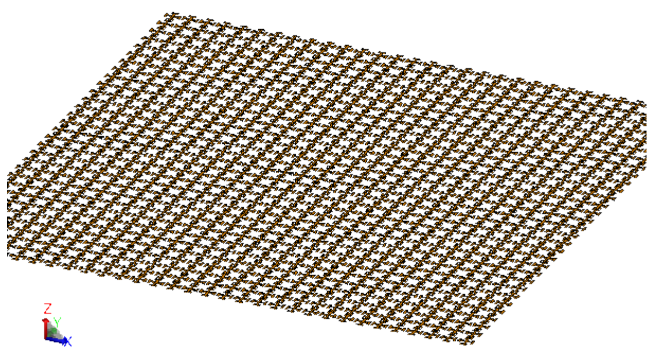


Figure 135: Full model of the 31x31 element FSS sheet solved with MLFMM.

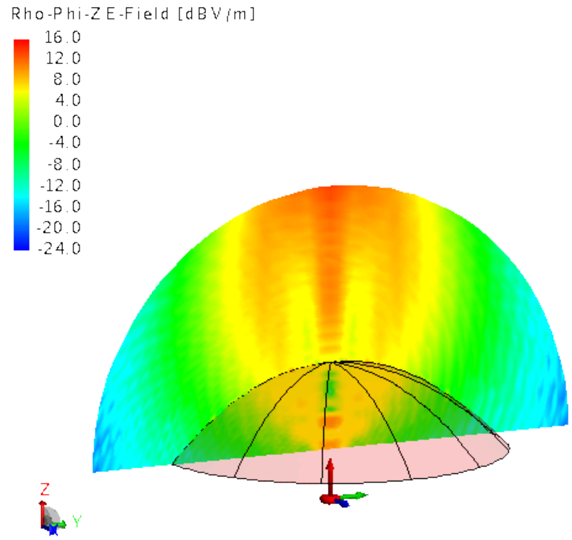


Figure 136: E-feld distribution through an FSS radome calculated using a characterised surfaces.

User Interface Tools

I

Simple examples demonstrating using Feko application automation, matching circuit generation with Optenni Lab and optimising a bandpass filter with HyperStudy.

This chapter covers the following:

I.1 Introduction to Application Automation

Use application automation to perform operations with CADFEKO and POSTFEKO. Typical tasks include repetitive tasks, tasks that require several steps, or calculations.

> **Note:** The base language for the application automation is Lua.
> - View the official Lua Reference Manual: http://www.lua.org/manual/5.1/
> - Programming introduction: http://www.lua.org/pil/
> - Community maintained Wiki: http://lua-users.org/wiki/

I.1.1 Automation Terminology

Learn the terminology of automation to create an automation script.

Lua	Lightweight multi-paradigm programming language designed as a scripting language. Both CADFEKO and POSTFEKO feature a scripting interface based on the Lua language.
API	The Feko application programming interface (API) defines the namespaces, objects, methods and functions used to access and control the Feko applications.
handle	A handle is a variable that refers to an object. Instead of referring to the object itself, the handle is used to refer to the object.
object	An object is an entity within an object oriented programming language with two main characteristics: a state and a behaviour. The settings of the object are stored in its properties and its abilities are accessed through methods.
`app`	The application is either CADFEKO or POSTFEKO.
`cf` / `pf` / `feko`	Each application has a "namespace" that groups all objects and properties for that particular application. The CADFEKO namespace is **cf** and the POSTFEKO namespace is **pf**. This means that all objects, static functions and nested namespaces will be accessed using the application's namespace. As an example, **`cf.GetApplication()`** calls the **`GetApplication()`** static function in the **`cf`** namespace. For functions that can be shared between the applications, a common namespace **`feko`** is also defined.
project	The project is the specific CADFEKO model the user is working on, for example, `patch_antenna.cfx`.

> **Note:** **`project`** is specific to CADFEKO.

type

Each object has a type. Lua supports the following standard types: number, boolean, string, table, function, userdata and nil. Objects created in CADFEKO or POSTFEKO will usually have a **`type`** property.

collection

A **`collection`** is a special object that contains objects of which there can be more than one. For example, there can be multiple sources, far fields, geometry parts and so on. When referencing an item in a collection, an index must always be specified, for example **`farfield[1]`** or **`farfield[`**"FarField"**`]`**.

enum

An enumerated list or **`enum`** is a set of options. In the graphical user interface these options relate to grouped options in a dropdown box or a radio button group. The user is unable to modify the list of options and must select one of the options.

Examples of enums:

- For the **Create dielectric medium** dialog select either the Frequency independent, Debye relaxation, Cole-Cole, Havriliak-Negami, Djordjevic-Sarkar or the Frequency list option for the Definition method.
- For the **Create mesh dialog**, select either the Fine, Standard, Coarse or Custom option for the **Mesh size**.

method

A **`method`** is a function that acts on a particular object. Methods are called using the ":" after the object to which the method belongs. For example, **`Cuboid:ConvertToPrimitive()`** returns the geometry in its primitive base form, whereas **`farField:Duplicate()`** returns a duplicate far field solution entity. **`Cuboid`** and **`farField`** are the objects and **`ConvertToPrimitive`** and **`Duplicate`** are the methods.

> **Note:** Use "." instead of ":" after **`cf`**.

static function

A **`static function`** is a function that is not associated with an object, as opposed to a method that works on a particular object. Static functions are called using ".".

> **Note:** Use "." instead of ":" after **`cf`**.

For example, **`cf.Cuboid.GetDefaultProperties()`** is a static function that creates the properties table for a cuboid.

`cf.GetApplication()` is a static function that returns the application object.

I.1.2 Navigate the API Documentation

Learn to navigate the API documentation to find the correct syntax for an automation script.

The application programming interface (API) documentation is contained in the Feko User Guide. The integrated help is simple to navigate due to its **Back** and **Forward** functionality and the **Index** or **Search** tab is useful to find a specific item.

For this example we will make use of the **Index** tab in the integrated help. For conciseness we will make use of the terminology, *search*, when we refer to the **Look for:** box on the **Index** tab.

Hyperlinks are indicated by blue text. Clicking on a hyperlink will navigate to other sections in the documentation.

The following example illustrates how to navigate the documentation and to use the correct syntax. The example contains steps to create a wire-fed patch antenna on a planar multilayer substrate.

> **Note:**
>
> The model is included in the Feko installation in the following directory:
>
> `examples\ExampleGuide_models\Example-I01-Introduction_to_Application_Automation`

I.1.3 Example: Patch Antenna on a Planar Multilayer Substrate

Learn to navigate the API documentation to create a patch antenna using an automation script.

The complete Lua script to create the model, is included in this example.

Creating a New Project

Define a new CADFEKO project.

```
myApplication = cf.Application.getInstance()
myProject = myApplication:NewProject()
```

1. Get a “handle” on the CADFEKO application.

   ```
   myApplication = cf.Application.getInstance()
   ```

2. Start a new empty project and get a “handle” on the project.

   ```
   myApplication:NewProject()
   ```

Defining a Point

Define a point at (-0.25, -0.25, 0) that will be used as the base corner of a patch (rectangle).

```
myBaseCorner = cf.Point(-0.25, -0.25, 0)
```

1. Search for *Point*[6] in the Help[7].
2. View the example and note that a point is defined as follows:

   ```
   cf.Point(y, y, z)
   ```

3. Fill in the coordinates (-0.25, -0.25, 0):

   ```
   cf.Point(-0.25, -0.25, 0)
   ```

4. Add a "handle" to the point:

   ```
   myBaseCorner = cf.Point(-0.25, -0.25, 0)
   ```

6. *Point* (Object)
7. Feko Scripting and API Reference Guide or WebHelp.

Creating a Patch (Rectangle)

Create a rectangle with base corner (*myBaseCorner*), *width* = 0.5, *depth* = 0.5 and label *Patch*.

```
myPatch = myProject.Contents.Geometry:AddRectangle(myBaseCorner, 0.5, 0.5)
myPatch.Label = "Patch"
```

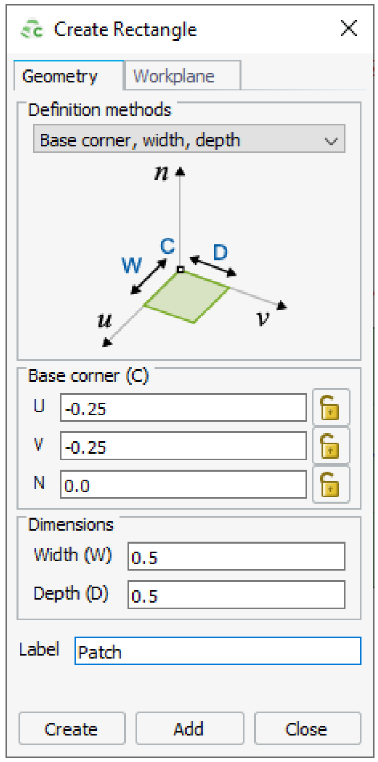


Figure 137: A rectangle with base corner (-0.25, -0.25, 0), width = 0.5, depth = 0.5 and label Patch.

1. A rectangle is a geometry object and since there may be multiple geometry objects in the model, it is part of the *GeometryCollection*.
2. Search for *GeometryCollection* in the Help[8].
3. In the Help, under **GeometryCollection** > **Method List**, search for applicable methods:
   - *AddRectangle (cornerpoint Point, width Expression, depth Expression)*
   - *AddRectangle (properties table)*
   - *AddRectangleAtCentre (centrepoint Point, width Expression, depth Expression)*

   To create a rectangle with a base corner, we will use the method:

   ```
   AddRectangle(cornerpoint Point, width Expression, depth Expression)
   ```
4. Fill in the corner point (use the point, *myBaseCorner*), *width* and *depth*:

   ```
   AddRectangle(myBaseCorner, 0.5, 0.5)
   ```

8. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

5. **Determine the syntax to prepend to** *AddRectangle*:
    a) Since *AddRectangle* is a method, it is indicated by prepending a ":" (colon).

    ```
    :AddRectangle(myBaseCorner, 0.5, 0.5)
    ```

    b) In the Help, under **GeometryCollection** > **Usage locations**, note the following:

    ```
    ModelContents object has collection Geometry[9]
    ```

    c) Click **ModelContents**.
    d) In the Help, under **ModelContents** > **Usage locations**, note the following:

    ```
    Model object has property Contents[10]
    ```

    Since we know that *Model* is the one of the top levels in the model tree, the result is then:

    ```
    Contents.Geometry:AddRectangle(myBaseCorner, 0.5, 0.5)
    ```

    e) Since the project is the highest level, we prepend our reference to the project:

    ```
    myProject.Contents.Geometry:AddRectangle(myBaseCorner, 0.5, 0.5)
    ```

6. Add a "handle" to the rectangle:

    ```
    myPatch = myProject.Contents.Geometry:AddRectangle(myBaseCorner, 0.5, 0.5)
    ```

7. Set the rectangle label to *Patch*:

    ```
    myPatch.Label = "Patch"
    ```

    Tip: View the *Rectangle* (object) in the Help for a short example.

9. The part that is prepended to the method, maps to the CADFEKO model tree structure.
10. The part that is prepended to the method, maps to the CADFEKO model tree structure.

Defining a Dielectric

Define the first dielectric with ε_r = 1.5 with label *Substrate1*.

```
myDiel1 = myProject.Definitions.Media.Dielectric:AddDielectric(1.5, 0, 1000)
myDiel1.Label = "Substrate1"
```

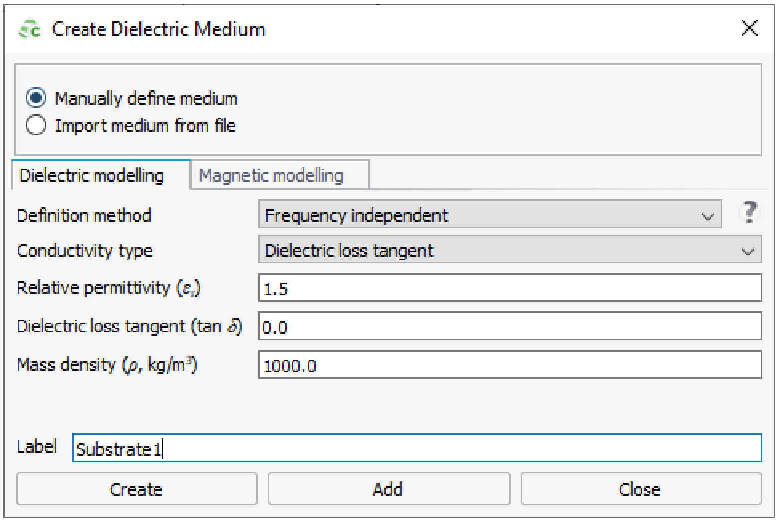


Figure 138: A dielectric with ε_r = 1.5 and label Substrate.

1. A dielectric is an object and since there may be multiple objects in the model, it is part of the *DielectricCollection*.
2. Search for *DielectricCollection* in the Help[11].
3. In **DielectricCollection**, under **Method List**, search for methods that are applicable to dielectrics:
   - *AddDielectric (properties table)*
   - *AddDielectric (relativepermittivity Expression, losstangent Expression, massdensity Expression)*
   - *AddDielectric ()*

   To create the first dielectric, we will use the method:

   ```
   AddDielectric(relativepermittivity Expression, losstangent Expression,
    massdensity Expression)
   ```
4. Fill in the values for the dielectric

   ```
   AddDielectric(1.5, 0, 1000)
   ```
5. Determine the syntax to prepend to *AddDielectric*:
   a) Since *AddDielectric* is a method, it is indicated by prepending a ":" (colon).

   ```
   :AddDielectric(1.5, 0, 1000)
   ```

11. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

b) In the Help, under **DielectricCollection** > **Usage locations**, note the following:

```
Media object has collection Dielectric[12]
```

Since we know that *Media* is the not one of the top levels in the model tree, click **Media**.

In the Help, under **Media** > **Usage locations** , note the following:

```
ModelDefinitions object has property Media.
```

c) Click **ModelDefinitions**.

d) In the Help, under **ModelDefinitions** > **Usage locations**, note the following:

```
Model object has property Definitions.
```

Since we know that **Model** is the one of the top levels in the model tree, the result is then:

```
Definitions.Media.Dielectric:AddDielectric(1.5, 0, 1000)
```

e) Since the project is the highest level, we prepend our reference to the project:

```
myProject.Definitions.Media.Dielectric:AddDielectric(1.5, 0, 1000)
```

6. Add a “handle” to the dielectric:

```
myDiel1 = myProject.Definitions.Media.Dielectric:AddDielectric(1.5, 0, 1000)
```

7. Set the dielectric label to *Substrate1*:

```
myDiel1.Label = "Substrate1"
```

> **Tip:** View the *Dielectric* (object) in the Help for a short example.

12. The part that is prepended to the method, maps to the CADFEKO model tree structure.

Defining a Dielectric Using the Properties Method

Define the second dielectric with ε_r = 2.5, σ = 1e-2 and with label *Substrate2*.

```
myProp = cf.Dielectric.GetDefaultProperties()
myDiel2 = myProject.Definitions.Media.Dielectric:AddDielectric(myProp)
myProp.MassDensity = "4"
myProp.Label = "Substrate2"
myProp.DielectricModelling.RelativePermittivity = "2.5"
myProp.DielectricModelling.ConductivityType
 = cf.Enums.MediumDielectricConductivityTypeEnum.Conductivity
myProp.DielectricModelling.Conductivity = "1e-2"
myDiel2:SetProperties(myProp)
```

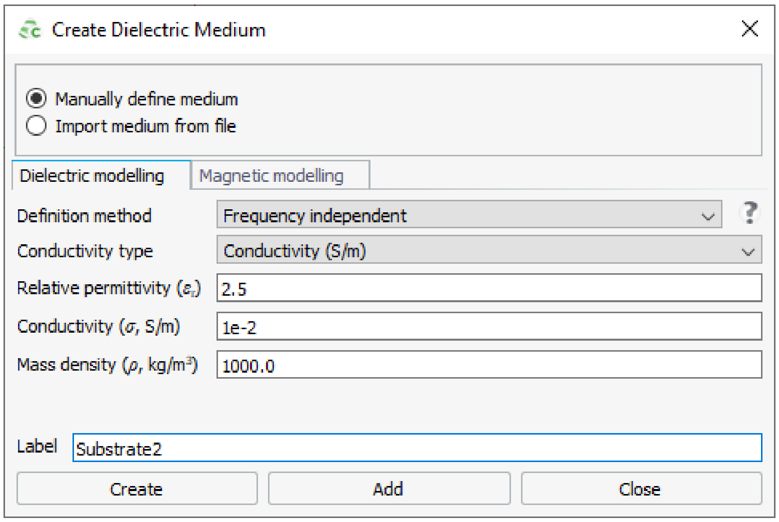


Figure 139: A dielectric with ε_r = 2.5, σ = 1e-2 and label Substrate2.

1. Create the second dielectric using the properties method:

```
AddDielectric (properties table)
```

2. Since we want to know the properties for a *Dielectric*, search for *Dielectric* (object) in the Help[13].
3. In the Help, under **Dielectric** > **Static Function List**, note the following:

```
GetDefaultProperties ()
```

4. Use `GetDefaultProperties()` to obtain the default properties of a dielectric:

```
myProperties = cf.Dielectric.GetDefaultProperties()
```

5. Specify the properties of the dielectric:
    a) In the Help, under **Dielectric** > **Property List**, note the properties of interest:
        - *Label*
        - *DielectricModelling*

13. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

Set the *Label* property, for *DielectricModelling* we need to navigate deeper:

```
myProperties.Label = "Substrate2"
```

b) In the Help, click on the link in **Dielectric** > **Static Function List** > **(Read/Write DielectricModelling)** to navigate to the dielectric modelling properties.

c) In the Help, under **DielectricModelling** > **Property List**, note the properties of interest:

- *RelativePermittivity*
- *Conductivity* and *ConductivityType*

Set the *RelativePermittivity* property, for *Conductivity* we need to navigate deeper:

```
myProperties.DielectricModelling.RelativePermittivity = "2.5"
```

d) In the Help, click on the link in **DielectricModelling** > **Property List** > **ConductivityType** > **MediumDielectricConductivityTypeEnum**.

e) Specify that conductivity will be used as well as the value for the conductivity:

```
myProperties.DielectricModelling.ConductivityType
 = cf.Enums.MediumDielectricConductivityTypeEnum.Conductivity
myProperties.DielectricModelling.Conductivity = "1e-2"
```

6. Update *myDielec1* with its new properties using `SetProperties ()`:

```
myDielec2:SetProperties(properties)
```

Creating a Planar Multilayer Substrate

Create a ground plane

```
properties =
 application.Project.Contents.SolutionSettings.GroundPlane:GetProperties()
properties.DefinitionMethod
 = cf.Enums.GroundPlaneDefinitionMethodEnum.MultilayerSubstrate
properties.Layers[1].GroundBottom = cf.Enums.GroundBottomTypeEnum.None
properties.Layers[1].Thickness = "0.351"
substrate2 = application.Project.Definitions.Media.Dielectric:Item("substrate2")
properties.Layers[1].Medium = substrate2
properties.Layers[2] = {}
properties.Layers[2].GroundBottom = cf.Enums.GroundBottomTypeEnum.PEC
properties.Layers[2].Thickness = "0.2"
substrate1 = application.Project.Definitions.Media.Dielectric:Item("substrate1")
properties.Layers[2].Medium = substrate1
properties.ZValue = "0.0"
application.Project.Contents.SolutionSettings.GroundPlane:SetProperties(properties)
```

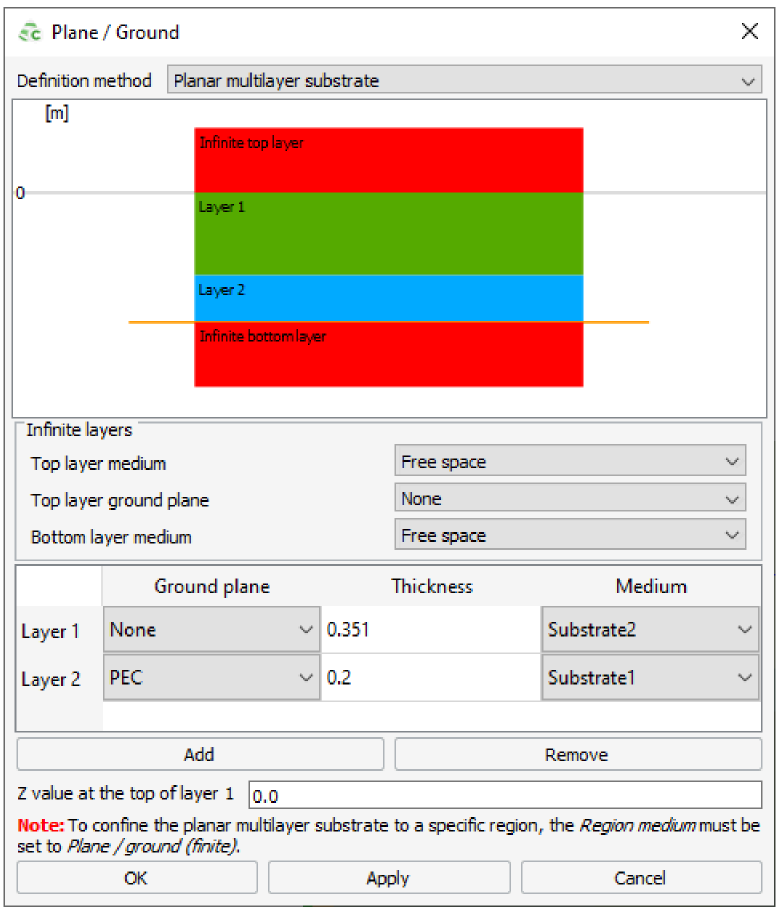


Figure 140: Define a planar multilayer substrate with tow layers. Layer 1 is 0.351

1. Since the *GroundPlane* object defines an infinite plane or ground, search for *GroundPlane* (object) in the Help[14].

14. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

a) In the Help, under **Launcher** > **Usage locations**, note the following:

```
SolutionSettings object has property GroundPlane.
```

2. In the Help, click on the link in **GroundPlane** > **Property List** > **DefinitionMethod** > **(Read/Write GroundPlaneDefinitionMethodEnum)** to specify the ground plane type.
3. *GroundPlaneDefinitionMethodEnum:*

a) In the Help, under **GroundPlaneDefinitionMethodEnum**, note the option:

```
MultilayerSubstrate
```

The result is then:

```
cf.Enums.GroundPlaneDefinitionMethodEnum = MultilayerSubstrate
```

Creating the Feed Line

Create a feed line with start point (0, 0, 0) , end point (0, 0, -0.551) and label *Feedline*.

```
myPoint1 = cf.Point(0, 0, 0)
myPoint2 = cf.Point(0, 0, -0.551)
myLine = myProject.Contents.Geometry:AddLine(myPoint1, myPoint2)
myLine.Label = "Feedline"
```

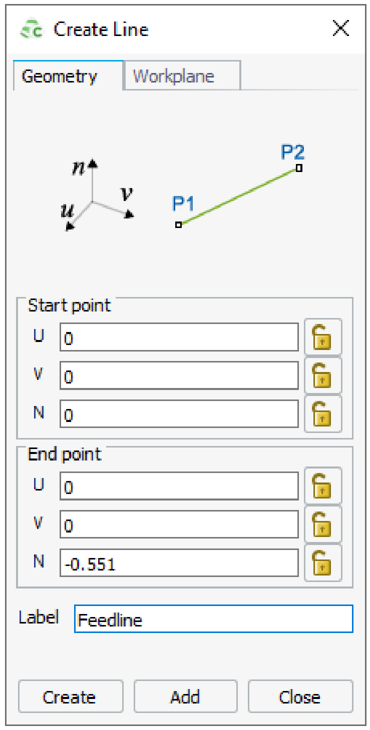


Figure 141: A line with start point (0, 0, 0) and end point (0, 0, -0.551).

1. Define two points, *myPoint1* and *mypoint2* (see Defining a Point).

```
myPoint1 = cf.Point(0, 0, 0)
myPoint2 = cf.Point(0, 0, -0.551)
```

2. A line is a geometry object and since there may be multiple geometry objects in the model, it is part of the *GeometryCollection*.
3. Search for *GeometryCollection* in the Help[15].
4. In the Help, under **GeometryCollection** > **Method List**, search for methods that are applicable to lines:
   - *AddLine (properties table)*
   - *AddLine (startpoint Point, endpoint Point)*

15. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

To create a line, we will use the method:

```
AddLine(startpoint Point, endpoint Point)
```

5. Fill in the *startpoint* and *endpoint* (use the points, *myPoint1* and *myPoint2*):

```
AddLine(myPoint1, myPoint2)
```

6. Determine the syntax to prepend to *AddLine*:

a) Since *AddLine* is a method, it is indicated by prepending a ":" (colon).

```
:AddLine(myPoint1, myPoint2)
```

b) In the Help, under **GeometryCollection** > **Usage locations**, note the following:

> **ModelContents** object has collection **Geometry**.[16]

c) Click **ModelContents**.

d) In the Help, under **ModelContents** > **Usage locations**, note the following:

> **Model** object has property **Contents**[17]

Since we know that *Model* is the one of the top levels in the model tree, the result is then:

```
Contents.Geometry:AddLine(myPoint1, myPoint2)
```

e) Since the project is the highest level, we prepend our reference to the project:

```
myProject.Contents.Geometry:AddLine(myPoint1, myPoint2)
```

7. Add a reference to the newly created line:

```
myLine = myProject.Contents.Geometry:AddLine(myPoint1, myPoint2)
```

8. Set the line label to *Feedline*:

```
myLine.Label = "Feedline"
```

> **Tip:** View the *Line* (object) in the Help for a short example.

16. The part that is prepended to the method, maps to the CADFEKO model tree structure.
17. The part that is prepended to the method, maps to the CADFEKO model tree structure.

Unioning the Model

Union all geometry in the model (*Feedline* and *Patch*) and set its label to *Union1*.

```
myUnion = myProject.Contents.Geometry:Union()
myUnion.Label = "Union1"
```

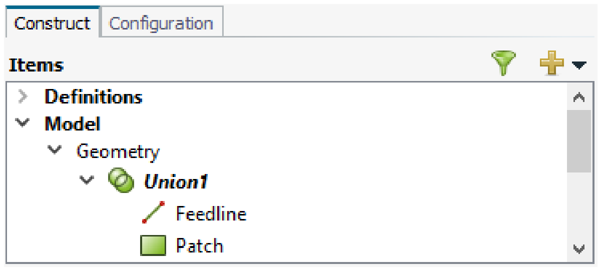


Figure 142: The model tree showing the union of Feedline and Patch with label Union1.

1. The union operation is a geometry object and since there may be multiple geometry objects in the model, it is part of the *GeometryCollection*.
2. Search for *GeometryCollection* in the Help[18].
3. In the Help, under **GeometryCollection** > **Method List**, search for methods that are applicable to the union operation:
   - *Union (geometrylist List of Geometry)*
   - *Union ()*

   To union all geometry in the model, we will use the method:

   ```
   Union()
   ```
4. Determine the syntax to prepend to *Union()*:
   a) Since *Union* is a method, it is indicated by prepending a ":" (colon).

   ```
   :Union
   ```

   b) In the Help, under **GeometryCollection** > **Usage locations**, note the following:

   ```
   ModelContents object has collection Geometry.[19]
   ```

   c) Click **ModelContents**.

   d) In the Help, under **ModelContents** > **Usage locations**, note the following:

   ```
   Model object has property Contents[20]
   ```

18. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.
19. The part that is prepended to the method, maps to the CADFEKO model tree structure.
20. The part that is prepended to the method, maps to the CADFEKO model tree structure.

Since we know that *Model* is the one of the top levels in the model tree, the result is then:

```
Contents.Geometry:Union()
```

e) Since the project is the highest level, we prepend our reference to the project:

```
myProject.Contents.Geometry:Union()
```

5. Add a reference to the newly created union:

```
myUnion = myProject.Contents.Geometry:Union()
```

6. Set the union label to *Union1*:

```
myUnion.Label = "Union1"
```

Tip: View the *Union* (object) in the Help for a short example.

Creating the Wire Port

Define a wire port at the end of *Feedline* with label *Port1*.

```
myPort = myProject.Contents.Ports:AddWirePort(myUnion.Wires[1])
myPort.Label = "Port1"
```

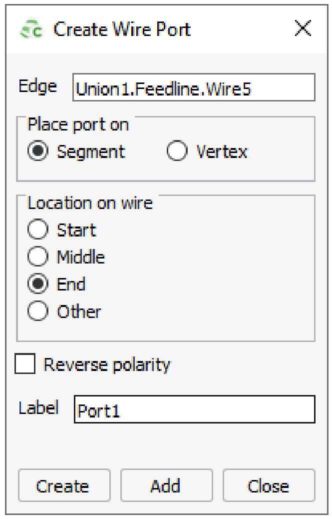


Figure 143: A wire port is placed at the end of Feedline with label Port1.

1. A port is a object and since there may be multiple objects in the model, it is part of the *PortCollection*.
2. Search for *PortCollection* in the Help[21].
3. In the Help, under **PortCollection** > **Method List**, search for methods that are applicable to geometry wire ports:
   - *AddWirePort (table table)*
   - *AddWirePort (wire Edge)*

   To create a wire port, we will use the method:

   ```
   AddWirePort(wire Edge)
   ```
4. Add the wire where wire port will be placed.

   ```
   AddWirePort(myUnion.Wires[1])
   ```

   Note: Use indexing to access a face, wire, edge or region in the details tree.
5. Determine the syntax to prepend to *AddWirePort*:
   a) Since *AddWirePort* is a method, it is indicated by prepending a ":" (colon).

   ```
   :AddWirePort(myUnion.Wires[1])
   ```

21. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

b) In the Help, under **PortCollection** > **Usage locations**, note the following:

```
ModelContents object has collection Ports.[22]
```

c) Click **ModelContents**.

d) In the Help, under **ModelContents** > **Usage locations**, note the following:

```
Model object has property Contents[23]
```

Since we know that *Model* is the one of the top levels in the model tree, the result is then:

```
Contents.Ports:AddWirePort(myUnion.Wires[1])
```

e) Since the project is the highest level, we prepend our reference to the project:

```
myProject.Contents.Ports:AddWirePort(myUnion.Wires[1])
```

6. Add a reference to the newly created wire port:

```
myPort = myProject.Contents.Ports:AddWirePort(myUnion.Wires[1])
```

7. Set the wire port label to *Port1*:

```
myPort.Label = "Port1"
```

Tip: View the *WirePort* (object) in the Help for a short example.

22. The part that is prepended to the method, maps to the CADFEKO model tree structure.
23. The part that is prepended to the method, maps to the CADFEKO model tree structure.

Creating a Voltage Source

Add a global voltage source to *Port1* for the default configuration, *StandardConfiguration1*.

```
myVoltageSource =
 myProject.Contents.SolutionConfigurations.GlobalSources:AddVoltageSource(myPort)
myVoltageSource.Label = "Source1"
```

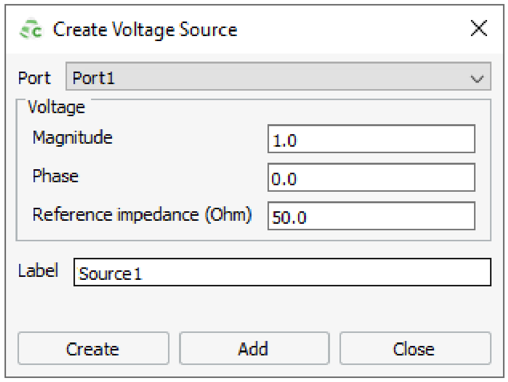


Figure 144: A voltage source is added to Port1.

1. A voltage source is a object and since there may be multiple objects in the model, it is part of the *SourceCollection*.
2. Search for *SourceCollection* in the Help[24].
3. In the Help, under **SourceCollection** > **Method List**, search for methods that are applicable to voltage sources:
   - *AddVoltageSource (properties table)*
   - *AddVoltageSource (portterminal Port)*

   To create a voltage source, we will use the method:

   ```
   AddVoltageSource(portterminal Port)
   ```
4. Fill in the port terminal (use port, *Port1*):

   ```
   AddVoltageSource(Port1)
   ```
5. Determine the syntax to prepend to *AddVoltageSource*:
   a) Since *AddVoltageSource* is a method, it is indicated by prepending a ":" (colon).

   ```
   :AddVoltageSource(myPort)
   ```
   b) In the Help, under **SoureCollection** > **Usage locations**, note the following:

   SolutionConfigurations collection has collection **GlobalSources**[25]

24. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.
25. The part that is prepended to the method, maps to the CADFEKO model tree structure.

Since we know that *Contents* is the one of the top levels in the configuration tree, the result is then:

```
Contents.SolutionConfigurations.GlobalSources:AddVoltageSource(myPort)
```

c) Since the project is the highest level, we prepend our reference to the project:

```
myProject.Contents.SolutionConfigurations.GlobalSources:AddVoltageSource(myPort)
```

6. Add a reference to the newly created voltage source:

```
myVoltageSource =
 myProject.Contents.SolutionConfigurations.GlobalSources:AddVoltageSource(myPort)
```

7. Set the voltage source label to *Source1*:

```
myPatch.Label = "Source1"
```

Tip: View the *VoltageSource* (object) in the Help for a short example.

Setting the Frequency

Specify a single global frequency as 300 MHz.

```
myProject.Contents.SolutionConfigurations.GlobalFrequency.Start = "300e6"
```

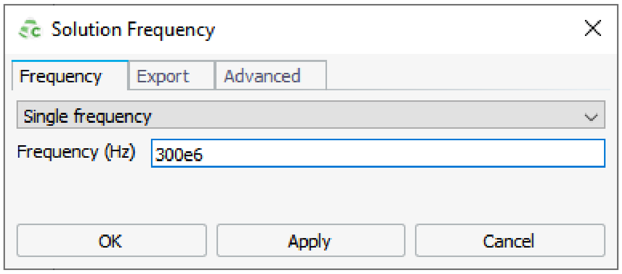


Figure 145: Specify a single global frequency as 300 MHz.

1. Search for *Frequency* (object) in the Help[26].
2. In the Help, under **Frequency** > **Property List**, search for properties that are applicable to setting a single frequency:
   - *Start* - The first frequency value (Hz).
3. Fill in the frequency value:

   ```
   Start = "300e6"
   ```

4. Determine the syntax to prepend to *Start*:
   a) Since *Start* is a property it is indicated by prepending a ".":

   ```
   .Start = "300e6"
   ```

   Tip: For a single frequency solution, you only need to specify the start frequency.

   b) In the Help, under **Frequency** > **Usage locations**, note the following:

   ```
   SolutionConfiguration object has property GlobalFrequency.
   ```

   Since *Contents* is the one of the top levels in the configuration tree, the result is:

   ```
   Contents.SolutionConfigurations.GlobalFrequency.Start = "300e6"
   ```

   c) Since the project is the highest level, we prepend our reference to the project:

   ```
   myProject.Contents.SolutionConfigurations.GlobalFrequency.Start = "300e6"
   ```

26. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

Specifying the Wire Segment Radius

Specify the wire segment radius as 0.001 m.

```
myProject.Mesher.Settings.WireRadius = "0.001"
```

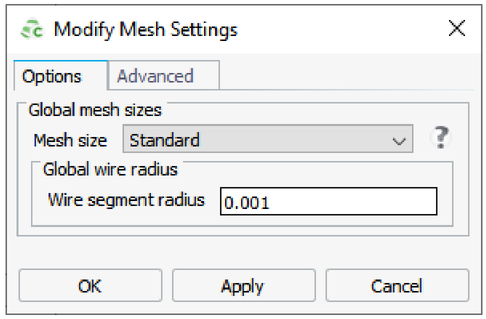


Figure 146: Specify a wire segment radius of 0.001 m

1. Since the *Mesher* object meshes the model, search for *Mesher* (object) in the Help[27].
   a) In the Help, under **Mesher** > **Usage locations** , note the following:

   ```
   Model object has property Mesher.
   ```

   b) In the Help, click on the link in **Mesher** > **Property List** > **Settings** > **(Read only MeshSettings)** to navigate to the properties applicable to mesh creation.
2. *MeshSettings* (object):
   a) In the Help, under **MeshSettings** > **Property List**, note the properties:
      - *WireRadius*
3. Fill in the wire segment radius:

   ```
   WireRadius = "0.001"
   ```

4. Determine the syntax to prepend to *WireRadius*:
   a) Since *Start* is a property it is indicated by prepending a ".":

   ```
   .WireRadius = "0.001"
   ```

   b) From 1.a and since we know that *Model* is the one of the top levels in the model tree, the result is then:

   ```
   Mesher.Settings.WireRadius = "0.001"
   ```

   c) Since the project is the highest level, we prepend our reference to the project:

   ```
   myProject.Mesher.Settings.WireRadius = "0.001"
   ```

27. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

Meshing the Model

Create a mesh.

```
myProject.Mesher:Mesh()
```

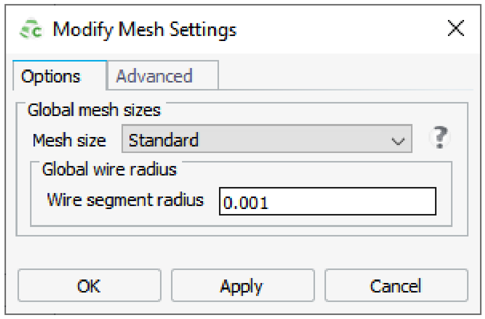


Figure 147: Specify a wire segment radius of 0.001 m

1. Since the *Mesher* object meshes the model, search for *Mesher* (object) in the Help[28].
   a) In the Help, under **Mesher** > **Usage locations**, note the following:

   ```
   Model object has property Mesher.
   ```

   b) In the Help, under **Mesher** > **Property List**, note the method:
   - *Mesh()*

2. Determine the syntax to prepend to *Mesh*:
   a) Since *Start* is a property it is indicated by prepending a ".":

   ```
   :Mesh()
   ```

   b) In the Help, under **MeshSettings** > **Usage locations**, note the following:

   ```
   Mesher object has property Settings.
   ```

   The result is then:

   ```
   myProject.Mesher:Mesh()
   ```

28. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

Defining a Far Field Request

Create a far field request (0°≤θ≤90°, with 0°≤ϕ≤360°). Sample the far field at θ=5° and ϕ=5° steps. The far field request is added to the default configuration, *StandardConfiguration1*.

```
myFarField = myProject.Contents.SolutionConfigurations[1].FarFields:Add(0, 0, 90, 360, 5, 5)
myFarField.Label = "FarField1"
```

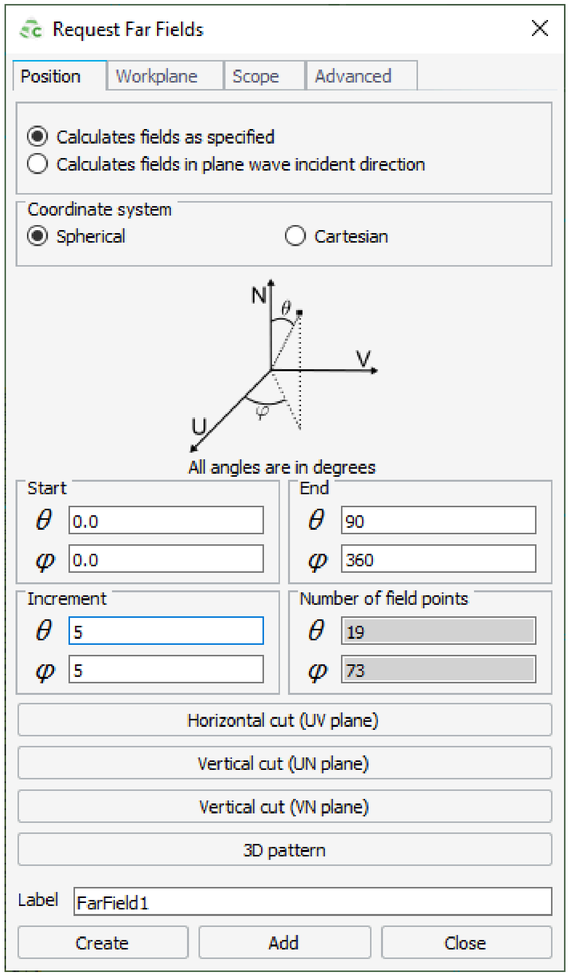


Figure 148: A far field request is added to the model.

1. A far field request is a object and since there may be multiple objects in the model, it is part of the *FarFieldCollection*.
2. Search for *FarFieldCollection* in the Help[29].
3. In the Help, under **FarFieldCollection** > **Method List**, search for methods that are applicable to adding a fra field request:
    - *Add (properties table)*
    - *Add (starttheta Expression, startphi Expression, endtheta Expression, endphi Expression, thetaincrement Expression, phiincrement Expression)*

29. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

- *Add3DPattern ()*
- *AddHorizontalCutUVPlane ()*
- *AddRequestInPlaneWaveIncidentDirection ()*
- *AddSquareGrid ()*
- *AddVerticalCutUNPlane ()*
- *AddVerticalCutVNPlane ()*

To create a far field request, we will use the method:

```
Add(starttheta Expression, startphi Expression, endtheta Expression, endphi
 Expression, thetaincrement Expression, phiincrement Expression)
```

4. Fill in the θ and ϕ values and increments:

```
Add(0, 0, 90, 360, 5, 5)
```

5. Determine the syntax to prepend to *Add*:
   a) Since *Add* is a method, it is indicated by prepending a ":" (colon).

   ```
   :Add(0, 0, 90, 360, 5, 5)
   ```

   b) In the Help, under **FarFieldCollection** > **Usage locations (collections)**, note that the following objects contain the **FarFieldCollection** collection:

   ```
   SolutionConfigurations(.FarFields)[30]
   ```

   Since we know that *Contents* is one of the top levels in the configuration tree, the result is:

   ```
   Contents.SolutionConfigurations.FarFields:Add(0, 0, 90, 360, 5, 5)
   ```

   But we also know that a far field request is added per configuration, the result is then:

   ```
   Contents.SolutionConfigurations[1].FarFields:Add(0, 0, 90, 360, 5, 5)
   ```

   c) Since the project is the highest level, we prepend our reference to the project:

   ```
   myProject.Contents.SolutionConfigurations[1].FarFields:Add(0, 0, 90, 360, 5, 5)
   ```

6. Add a reference to the newly created far field request:

```
myFarField =
 myProject.Contents.SolutionConfigurations[1].FarFields:Add(0, 0, 90, 360, 5, 5)
```

7. Set the far field request label to *Source1*:

```
myFarField.Label = "FarField1"
```

Tip: View the *FarField* (object) in the Help for a short example.

30. The part that is prepended to the method, maps to the CADFEKO model tree structure.

Defining a Current Request

Create a currents request. The currents request is added for the default configuration, *StandardConfiguration1*.

```
myCurrents = myProject.Contents.SolutionConfigurations[1].Currents:Add()
myCurrents.Label = "Currents"
```

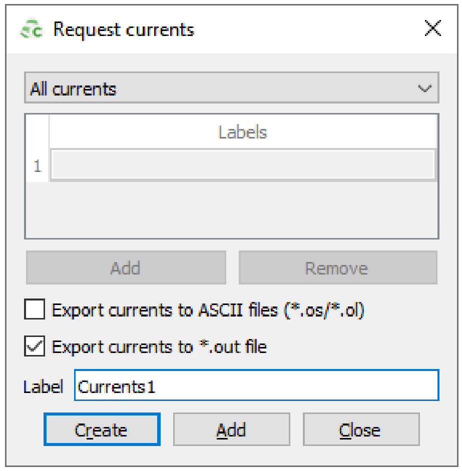


Figure 149: A current request is added to the model.

1. A currents request is a object and since there may be multiple objects in the model, it is part of the *CurrentsCollection*.
2. Search for *CurrentsCollection* in the Help[31].
3. In the Help, under **CurrentsCollection** > **Method List**, search for methods that are applicable to adding a currents request:
   - *Add (properties table)*
   - *Add ()*

   To create a currents request, we will use the method:

   ```
   Add ()
   ```
4. Determine the syntax to prepend to *Add*:
   a) Since *Add()* is a method, it is indicated by prepending a ":" (colon).

   ```
   :Add()
   ```

   b) In the Help, under **CurrentsCollection** > **Usage locations (collections)**, note that the following objects contain the **CurrentsCollection** collection:

   ```
   SolutionConfigurations(.Currents)[32]
   ```

31. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.
32. The part that is prepended to the method, maps to the CADFEKO model tree structure.

Since we know that *Contents* is the one of the top levels in the configuration tree, the result is then:

```
Contents.SolutionConfigurations.Currents:Add()
```

But we also know that a currents request is added per configuration, the result is then:

```
Contents.SolutionConfigurations[1].Currents:Add()
```

c) Since the project is the highest level, we prepend our reference to the project:

```
myProject.Contents.SolutionConfigurations[1].Currents:Add()
```

5. Add a reference to the newly created currents request:

```
myCurrents = myProject.Contents.SolutionConfigurations[1].Currents:Add()
```

6. Set the far field request label to *Currents1*:

```
myCurrents.Label = "Currents"
```

Tip: View the *Currents* (object) in the Help for a short example.

Saving the Project

Save the project to a `.cfx` file.

```
myApplication:SaveAs("Example-I01-Introduction_to_Application_Automation.cfx")
```

1. Since all actions like loading a project, saving a project or exiting the application are done on the application level, search for *application* (object) in the Help[33].
2. In the Help, under **Application** > **Method List**, search for methods that are applicable to saving the project:
   - *Save ()*
   - *SaveAs (filename string)*

   To specify a file name and save the model, we will use the method:

   ```
   SaveAs (filename string)
   ```
3. Fill in the file name:

   ```
   SaveAs("Example-I01-Introduction_to_Application_Automation.cfx")
   ```
4. Determine the syntax to prepend to *SaveAs()*:
   a) Since *SaveAs* is a method, it is indicated by prepending a ":" (colon).

   ```
   :SaveAs("Example-I01-Introduction_to_Application_Automation.cfx")
   ```
   b) Since *SaveAs* is a method on the *Application* object, prepend our reference to *Application*, *myApplication*:

   ```
   myApplication:SaveAs("Example-I01-Introduction_to_Application_Automation.cfx")
   ```

33. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

Enabling the Parallel Solver

Activate the parallel solver and specify the number of parallel processes as four.

```
myApplication.Launcher.Settings.FEKO.Parallel.NumberOfProcessesEnabled = true
myApplication.Launcher.Settings.FEKO.Parallel.ProcessCount = 4
```

1. Since the *Launcher* object coordinates the launching of Feko and external processes, search for *Launcher* (object) in the Help[34].
   a) In the Help, under **Launcher** > **Usage locations (object properties)**, note that the following objects have properties using the *Launcher* object:

   ```
   Application(.Launcher)
   ```

   b) In the Help, click on the link in **Launcher** > **Property List** > **Settings** > **(Read/Write ComponentLaunchOptions)** to navigate to the component launch options properties.
2. *ComponentLaunchOptions* (object):
   a) In the Help, under **ComponentLaunchOptions** > **Usage locations (object properties)**, note that the following objects have properties using the *ComponentLaunchOptions* object:

   ```
   Launcher(.Settings)
   ```

   b) In the Help, click on the link in **ComponentLaunchOptions** > **Property List** > **FEKO** > **(Read/Write FEKOLaunchOptions)** to navigate to the Feko launch options.
3. *FEKOLaunchOptions (Object):*
   a) In the Help, under **FEKOLaunchOptions** > **Usage locations (object properties)**, note that the following objects have properties using the *FEKOLaunchOptions* object:

   ```
   Settings(.FEKO)
   ```

   b) In the Help, click on the link in **FekoLaunchOptions** > **Property List** > **Parallel** > **(Read/Write FEKOParallelExeccutionOptions)** to navigate to the parallel execution options.
4. *FEKOParallelExecutionOptions* (object):
   a) In the Help, under **FEKOParallelExecutionOptions** > **Usage locations (object properties)**, note that the following objects have properties using the *FEKOParallelExecutionOptions* object:

   ```
   FEKO(.Parallel)
   ```

   b) In the Help, under **FEKOParallelExecutionOptions** > **Property List**, note the properties:
   - *NumberOfProcessesEnabled*
   - *ProcessCount*

   The result is then:

   ```
   myApplication.Launcher.Settings.FEKO.Parallel.NumberOfProcessesEnabled = true
   myApplication.Launcher.Settings.FEKO.Parallel.ProcessCount = 4
   ```

34. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

Running the Solver

Launch the Solver.

```
myApplication.Launcher:RunFEKO()
```

1. Since the *Launcher* object coordinates the launching of Feko and external processes, search for *Launcher* (object) in the Help[35].
2. In the Help, under **Launcher** > **Method List**, search for a method to launch the Solver:
   - *RunFEKO()*
3. Determine the syntax to prepend to *RunFEKO()*:
   a) Since *RunFEKO()* is a method, it is indicated by prepending a ":" (colon):

   ```
   :RunFEKO()
   ```

   b) In the Help, under **Launcher** > **Usage locations (object properties)**, note that the following objects have properties using the *Launcher* object:

   ```
   Application(.Launcher)
   ```

   The result is then:

   ```
   myApplication.Launcher:RunFEKO
   ```

35. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

I.1.4 Example: Post-Processing the Results

Post-process the results of a patch antenna on a multilayer substrate.

Learn to navigate the API documentation to do post-processing with POSTFEKO.

Creating a New POSTFEKO Project

Define a new POSTFEKO project and open a `.fek` file.

```
myApplication = pf.GetApplication()
myApplication:NewProject()
myApplication:OpenFile("Example-I01-Introduction_to_Application_Automation.fek")
```

1. Get a “handle” on the POSTFEKO application.

   ```
   myApplication = pf.GetApplication()
   ```

2. Start a new empty project.

   ```
   myApplication:NewProject()
   ```

3. Open the `.fek` file of the patch antenna already solved.

   ```
   myApplication:OpenFile("Example-I01-Introduction_to_Application_Automation.fek")
   ```

Getting a "Handle" on the Far Field Result

Get a "handle" on the far field result in the far field collection.

```
myFarFieldResult = myApplication.Models[1].Configurations[1].FarFields[1]
```

1. A far field result is an object and since there may be multiple far field results in a project, it is part of the **FarFieldCollection**.
2. Search for **FarFieldCollection** in the Help[36].
3. In **FarFieldCollection**, under **Index List**, note the following options to specify a specific far field result in the collection:
   - *[number]*
   - *[string]*

   To specify the far field data in the collection, we will use *[number]* since we only added a single far field request (as a result there will only be one far field result.
4. In the Help, under **FarFieldCollection** > **Usage locations**, note the following:

   ```
   SolutionConfiguration object has collection FarFields.
   ```

   The result is then:

   ```
   FarFields[1]
   ```
5. Determine the syntax to prepend to *FarFields[1]*:
   a) Since we know that far field requests are defined per configuration and there may be multiple configurations in the project, it is part of the **ConfigurationCollection**.
   b) In the Help, under **ConfigurationCollection** > **Usage locations**, note the following:

   ```
   Model object has collection Configurations.
   ```

   c) In **ConfigurationCollection**, under **Index List**, note the following options to specify a specify a configuration in the collection:
   - *[number]*
   - *[string]*

   To specify the configuration in the collection, we will use *[number]* since the model only contains a single configuration. The result is then:

   ```
   Configurations[1].FarFields[1]
   ```
6. Determine the syntax to prepend to *Configurations[1]*:
   a) Since we know that there may be multiple models in a project a, it is part of the **ModelCollection**.

36. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

b) In the Help, under **ModelCollection** > **Usage locations**, note the following:

```
Application object has collection Models.
```

c) In **ModelCollection**, under **Index List**, note the following options to specify a specific model in the collection:

- *[number]*
- *[string]*

To specify the model in the project, we will use *[number]* since the project only contains a single model. The result is then:

d) Since we already have a "handle" on the application, the result is:

```
myApplication.Models[1].Configurations[1].FarFields[1]
```

7. Add a reference to the far field result:

```
myFarFieldResult = myApplication.Models[1].Configurations[1].FarFields[1]
```

Adding a Far Field Result Trace to A Cartesian Graph

Create a Cartesian graph and add a far field result trace, change the independent and fixed axes, quantities and format the graph.

Adding a Cartesian Graph

Add a Cartesian graph to the project.

```
myGraph = myApplication.CartesianGraphs:Add()
```

1. A Cartesian graph is an object and since there may be multiple Cartesian graphs in the project, it is part of the *CartesianGraphCollection*.
2. Search for *CartesianGraphCollection* in the Help[37].
3. In **CartesianGraphCollection**, under **Method List**, search for an applicable method:
   - *Add()*
4. Determine the syntax to prepend to *Add()*:
   a) Since *Add()* is a method, it is indicated by prepending a ":" (colon).

   ```
   :Add()
   ```

   b) In the Help, under **CartesianGraphCollection** > **Usage locations**, note the following:

   ```
   Application object has collection CartesianGraphs.
   ```

   The result is then:

   ```
   CartesianGraphs:Add()
   ```

   c) Since we already have a "handle" on the application, the result is then:

   ```
   myApplication.CartesianGraphs:Add()
   ```

5. Add a reference to the newly created graph:

   ```
   myGraph = myApplication.CartesianGraphs:Add()
   ```

37. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

Adding a Result Trace to a Cartesian Graph

Add a trace (for this example, a far field result) to a Cartesian graph.

```
myFarFieldTrace = myGraph.Traces:Add(myFarFieldResult)
```

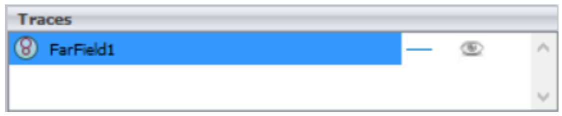


Figure 150: The traces panel in the result palette.

1. A trace is an object and since there may be multiple traces in the project, it is part of the **`ResultTraceCollection`**.
2. Search for **`ResultTraceCollection`** in the Help[38].
3. In **ResultTraceCollection**, under **Method List**, search for an applicable method:
   - *Add(result)*

   Since we already have a "handle" on the far field result, the result is:

   ```
   Add(myFarFieldResult)
   ```
4. Determine the syntax to prepend to *Add(myFarFieldResult)*:
   a) Since *Add(myFarFieldResult)* is a method, it is indicated by prepending a ":" (colon).

   ```
   :Add(myFarFieldResult)
   ```

   b) In the Help, under **ResultTraceCollection** > **Usage locations**, note the following:

   ```
   CartesianGraph object has collection Traces.
   ```

   The result is then:

   ```
   Traces:Add(myFarFieldResult)
   ```

   c) Since we already have a "handle" on the Cartesian graph, the result is:

   ```
   myGraph.Traces:Add(myFarFieldResult)
   ```
5. Add a reference to the trace:

   ```
   myFarFieldTrace = myGraph.Traces:Add(myFarFieldResult)
   ```

38. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

Changing the Axes of the Trace (Independent and Fixed)

Set the independent axis of the far field trace to Phi and the fixed axis to Theta = 30°.

```
myFarFieldTrace.IndependentAxis = "Phi"
myFarFieldTrace:SetFixedAxisValue("Theta",30, "deg")
```

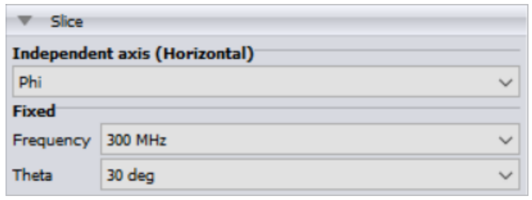


Figure 151: The Slice panel in the result palette.

1. Search for the **`FarFieldTrace`** object in the Help[39].
2. In the Help under **FarFieldTrace** > **Property List**, search for an applicable property to specify the independent axis:
   - *IndependentAxis*

   The result is then:

   ```
   IndependentAxis = "Phi"
   ```
3. Determine the syntax to prepend to *IndependentAxis*:
   a) Since *IndependentAxis* is a property, it is indicated by prepending a ".":

   The result is then:

   ```
   .IndependentAxis = "Phi"
   ```
   b) Since we already have a handle on the trace, the result is:

   ```
   myFarFieldTrace.IndependentAxis = "Phi"
   ```
4. In the Help under **FarFieldTrace** > **Method list**, search for an applicable method to specify the fixed axis:
   - *SetFixedAxisValue (axis string, numvalue number, unit string)*
   - *SetFixedAxisValue (axis string, strvalue string)*

   To specify the fixed axis, we will use the method:

   ```
   SetFixedAxisValue (axis string, numvalue number, unit string)
   ```

   The result is then:

   ```
   SetFixedAxisValue("Theta", 30, "deg")
   ```
5. Determine the syntax to prepend to *IndependentAxis*:

39. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

a) Since *SetFixedAxisValue* is a method, it is indicated by prepending a ":" (colon):

The result is then:

```
:SetFixedAxisValue("Theta", 30, "deg")
```

b) Since we already have a handle on the trace, the result is:

```
myFarFieldTrace:SetFixedAxisValue("Theta", 30, "deg")
```

Modifying the Quantities of a Far Field Result Trace

Change the far field trace axis to dB.

```
myFarFieldTrace.Quantity.ValuesScaledToDB = true
```

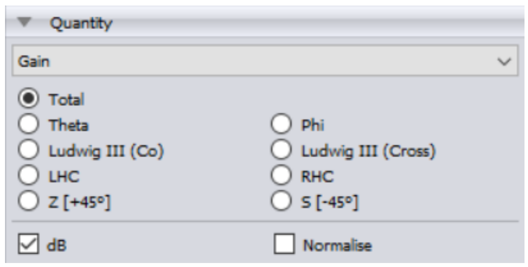


Figure 152: The quantity panel in the result palette.

1. Search for the `FarFieldQuantity` object in the Help[40].
2. In the Help under **FarFieldQuantity** > **Property List**, search for a property applicable to changing the trace values to dB:
   - *ValuesScaledToDB*
3. In the Help, under **FarFieldQuantity** > **Usage locations**, note the following:

   ```
   FarFieldTrace object has property Quantity.
   ```

   The result is then:

   ```
   Quantity.ValuesScaledToDB = true
   ```
4. Since we already has a "handle" on the far field trace, the result is:

   ```
   myFarFieldTrace.Quantity.ValuesScaledToDB = true
   ```

40. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

Formatting the Graph

Change the font size for both the vertical axis and horizontal axis to 12.

```
myGraph.HorizontalAxis.Title.Font.Size = 12
myGraph.VerticalAxis.Title.Font.Size = 12
```

1. Search for the **CartesianGraph** object in the Help[41].
2. In the Help under **CartesianGraph** > **Property List**, search for properties applicable to graph axes:
   - *HorizontalAxis*
   - *VerticalAxis*

   a) In the Help under **CartesianGraph** > **Property List**, click on the link **(Read only HorizontalAxis)** to navigate to the *HorizontalAxis* object.
3. In the Help under **HorizontalGraphAxis** > **Property List**, search for a property applicable to the axis title:
   - *Title*

   a) In the Help under **HorizontalGraphAxis** > **Property List**, click on the link **(Read only GraphAxisTitle)** to navigate to the *HorizontalAxis* object.
4. In the Help under **GraphAxisTitle** > **Property List**, search for a property applicable to font:
   - *Font*

   a) In the Help under **GraphAxisTitle** > **Property List**, click on the link **(Read only FontFormat)** to navigate to the *FontFormat* object.
5. In the Help under **FontFormat** > **Property List**, search for a property applicable to the axis title:
   - *Size*

   a) In the Help under **HorizontalGraphAxis** > **Property List**, click on the link **(Read only GraphAxisTitle)** to navigate to the *HorizontalAxis* object.

   The result is then:

   ```
   HorizontalAxis.Title.Font.Size
   ```
6. Since we already have a handle on the Cartesian graph, the result is then:

   ```
   myGraph.HorizontalAxis.Title.Font.Size
   ```
7. Specify the font size as 12.

   ```
   myGraph.HorizontalAxis.Title.Font.Size = 12
   ```
8. Similar to Step 7, the font size can be specified for the vertical axis:

   ```
   myGraph.VerticalAxis.Title.Font.Size = 12
   ```

41. The API is available in the Feko Scripting and API Reference Guide (PDF) or Feko WebHelp.

Script

View the completed POSTFEKO automation script.

The completed script.

```
--[[PATCH ANTENNA ON PLANAR MULTILAYER SUBSTRATE
===================================================================
This script post-process the calculated results of the patch antenna
placed on a planar multilayer substrate.
]]--
app = pf.GetApplication()
app:NewProject()
app:OpenFile("patch_antenna_scripted_model.fek")

-- Add a Cartesian graph
my_graph = app.CartesianGraphs:Add()

-- Add a far field result to a Cartesian graph
my_farfield_trace = my_graph.Traces:Add(app.Models[1].Configurations[1].FarFields[1])

-- Scale the quantity to dB
my_farfield_trace.Quantity.ValuesScaledToDB = true

-- Set the independent axis to Phi
my_farfield_trace.IndependentAxis = "phi"

-- Set the fixed axis to Theta = 30 deg
my_farfield_trace:SetFixedAxisValue("theta",30, "deg")

-- Add the surface currents to the 3D view
this_3Dview = app.Views[1]
my_3Dview_currents_plot =
 this_3Dview.Plots:Add(app.Models[1].Configurations[1].SurfaceCurrents[1])

-- Scale the quantity to dB
my_3Dview_currents_plot.Quantity.ValuesScaledToDB = true

-- Set the font size of the title
my_graph.VerticalAxis.Title.Font.Size = 14
my_graph.HorizontalAxis.Title.Font.Size = 14

-- Set the font size of the labels
my_graph.HorizontalAxis.Labels.Font.Size = 14
my_graph.VerticalAxis.Labels.Font.Size = 14
```

I.2 Automatic Report Generation Using API

Increase productivity when dealing with predictable and repeatable POSTFEKO sessions (for example, exporting a report) using application automation. Use an automation script to configure a session and export a report that highlights the antenna properties of the model.

> **Note:** The script can be used on various models with repeatable results.

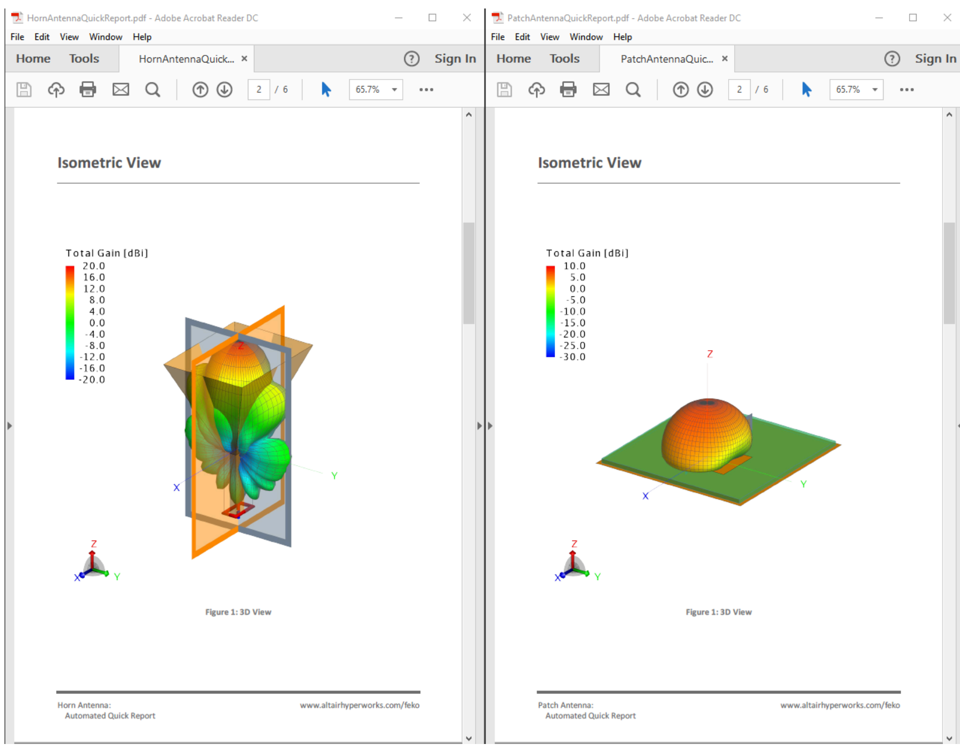


Figure 153: Results of the automatic report generation for two different antenna types.

I.2.1 The Models

Two models are included with the example to provide content to export a report.

The following assumptions are made for both models:

- The first configuration contains 3D far field data.
- The far field can be wrapped in the θ direction.
- There is a ϕ angle calculated at ϕ=0° and at ϕ=90°. This also implies that the main direction of radiation is in the positive Z axis.

> **Note:** Apart from these assumptions, any antenna geometry can be used as an input. The script can be adapted to iterate over multiple models or to display different properties of interest.

I.2.2 Script

View the script used to configure a session and export a report that highlights the antenna properties of the model.

```
--[[
AUTOMATIC QUICK REPORT GENERATION FOR ANTENNA PATTERN ANALYSIS
==============================================================
This script loads the specified model. It then creates various
views and graphs that display the far field pattern. These
views are then exported to a PDF report
]]--
modelName = {}
modelName[1] = "Horn"
modelName[2] = "Patch"
-- Get user input through Forms
form = pf.Form.New("Select model")
comboBox = pf.FormComboBox.New("Model name", modelName)
form:Add(comboBox)
form:Run()
index = comboBox.Index
app = pf.GetApplication()
app:NewProject()
app:OpenFile(modelName[index]..".fek")
selectedModel = app.Models[modelName[index]]
selectedConfig1 = selectedModel.Configurations[1]
ffData = selectedConfig1.FarFields[1] -- This is a handle on the far field data
 itself
view3D = app.Views[1]
ffPlot = view3D.Plots:Add(ffData)
ffPlot.Label = "ff3D"
ffPlot.Quantity.ValuesScaledToDB = true
view3D_top = view3D:Duplicate()
view3D_top:SetViewDirection(pf.Enums.ViewDirectionEnum.Top)
view3D_right = view3D:Duplicate()
view3D_right:SetViewDirection(pf.Enums.ViewDirectionEnum.Right)
view3D_front = view3D:Duplicate()
view3D_front:SetViewDirection(pf.Enums.ViewDirectionEnum.Front)
polarGraph = app.PolarGraphs:Add()
ffTracePhi_00 = polarGraph.Traces:Add(ffData)
ffTracePhi_00.IndependentAxis = "Theta (wrapped)"
ffTracePhi_00.Quantity.ValuesScaledToDB = true
ffTracePhi_90 = polarGraph.Traces:Add(ffData)
ffTracePhi_90.IndependentAxis = "Theta (wrapped)"
ffTracePhi_90.Quantity.ValuesScaledToDB = true
ffTracePhi_90:SetFixedAxisValue(ffTracePhi_90.FixedAxes[2],90,"deg")
polarGraph:ZoomToExtents()
polarGraph.Title.Text = "Gain"
polarGraph.Legend.Position = pf.Enums.GraphLegendPositionEnum.OverlayTopRight
polarGraph.BackColour = pf.Enums.ColourEnum.LightGrey
polarGraph:Restore()
quickReport = app:CreateQuickReport(modelName[index].."AntennaQuickReport",
pf.Enums.ReportDocumentTypeEnum.PDF)
quickReport.DocumentHeading = modelName[index]..[[ Antenna:
Automated Quick Report]]
quickReport:SetPageTitle(view3D.WindowTitle, "Isometric View")
quickReport:SetPageTitle(view3D_top.WindowTitle, "Top View")
quickReport:SetPageTitle(view3D_right.WindowTitle, "Right View")
quickReport:SetPageTitle(view3D_front.WindowTitle, "Front View")
quickReport:SetPageTitle(polarGraph.WindowTitle, "Theta Cuts")
```

```
quickReport:Generate()
```

I.2.3 Exercise 1

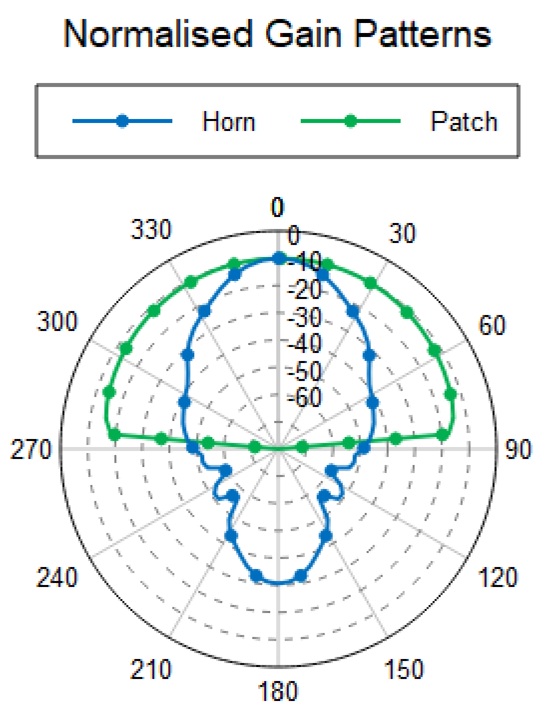
Modify the provided script to complete exercise 1.

- Change the rendering of the mesh to be 60% opaque.
- Automatically generate reports for all of the models in the *modelName* list.
- Save the sessions under unique names.

I.2.4 Exercise 2

Write a new script using the documentation to complete exercise 2.

1. Reproduce the following polar graph.



2. Export the graph to a `.pdf` file.
3. Save the session.

I.3 Using Altair HyperStudy with Feko to Optimise a Bandpass Filter

Design a bandpass coupled line filter with Altair HyperStudy as optimisation engine and Feko as the computational solver.

The design aims to obtain the best spacings for S1 - S3 in the depiction of the geometry below. The spacings are optimised to maximize the coupling between the input and output ports within the operating frequency range.

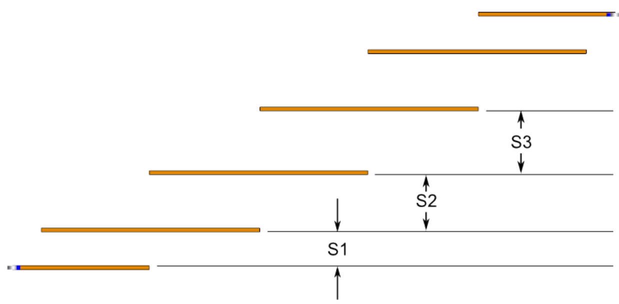


Figure 154: Top view of the bandpass filter.

> **Note:** The parametric model is provided for convenience. For the sake of simplicity, the cutoff frequencies and out-of-band performance will not be considered for the design.

I.3.1 HyperStudy Model Modification and Interpretation Approaches

Learn the four approaches in HyperStudy to modify a model and interpret the results.

The *design of experiments* (DOE) approach

The DOE approach can be defined as a test or a series of tests in which purposeful changes are made to the input variables of a process or system so that the reasons for changes in the output response can be identified and observed. Responses can be extracted, but will not affect how the model permutations are generated.

A *fit* approach

The fit approach approximates the response of a model by creating a mathematical equivalent of a model. This approach uses previous approaches as inputs that are used to predict how a model would behave for a change in design variables. This approach is recommended where computational resources are scarce.

The *optimisation* approach

Optimisation approaches are used to generate a model that behaves in the desired manner. The responses that are extracted from simulations are used to determine what the next model permutation should be.

The *stochastic* approach

Stochastic approaches are used to analyse the effect of tolerances in the design variables (for example, from material properties, manufacturing tolerances). These approaches can help identify the probability of responses adhering to defined specification.

I.3.2 Workflow

Learn the HyperStudy workflow. Learn study setup and configuration phases.

A typical workflow for HyperStudy is as follows:

1. Configure a study setup.

   This includes defining which models need to be included in a study, which solvers need to be used, which design variables may be altered and which responses to analyse.
2. Run the study by using any of the four approaches, "Design of Experiments", "Fit", "Stochastic" or "Optimisation."

   Feko is a registered solver for HyperStudy. When a solver is registered the majority of the workflow is integrated into Hyperstudy. The following phases form part of the configuration process.

   Define models

   With this phase, the solver input file is provided. HyperStudy will create this file for each run from the initial template file, using the current value of the design variables. Once the model has been added, the **Import Variables** button will analyse the model and identify design variables that may be modified. These variables can be used in subsequent approaches to generate different model permutations.

   Define design variables

   A table of variables have been imported in the model definition phase. These variables may now be edited, deactivated and the value ranges can be set.

   Evaluate

   This phase is responsible to **write** files, **execute** the solver and **extract** values from the results.

   - **Write**: During the write phase, HyperStudy will create subdirectories with a copy of the model and all of the files that the model depends on to be executed. The model variables will be updated before executing the solver.
   - **Execute**: During the execution phase, the solver will be run and output data will be generated. Note that this

includes any post-processing of the raw simulation data (e.g. by running a script on output data).

- **Extract**: The extraction phase is responsible for identifying output response values and pulling them back into HyperStudy.

> **Note:** ASCII files can be processed during the extraction phase.

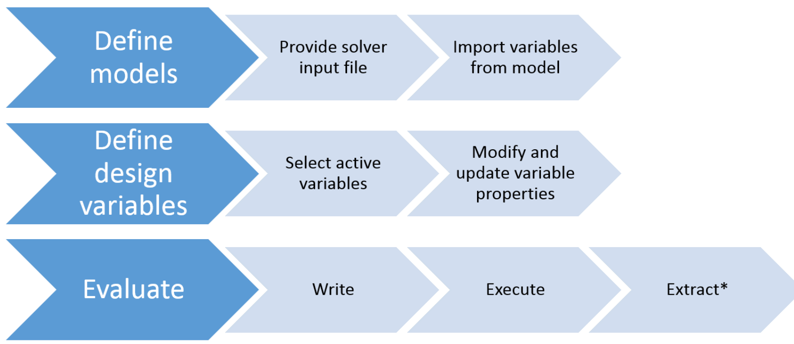
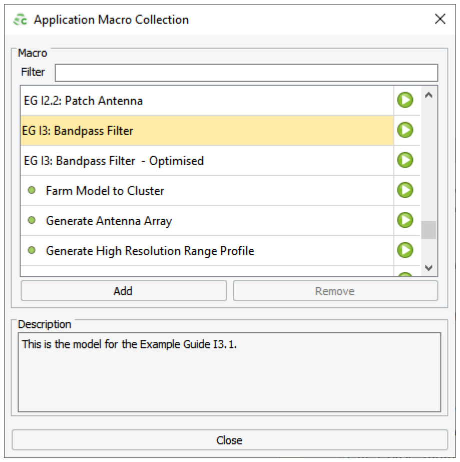


Figure 155: HyperStudy and Feko integration workflow.

I.3.3 Optimising the Bandpass Filter with HyperStudy Using a POSTFEKO Session

Configure the `bandpassfilter.pfs` file with the S11 data to minimise the reflection coefficient in HyperStudy.

1. Open CADFEKO.
2. Click **Application Macro** under the **Scripting** group.
3. In the drop-down list, select **Macro Library** and run the **EG I3: Bandpass Filter** script.



4. Save the model with the name *bandpassfilter.cfx* and execute the solver.
5. Open the *bandpassfilter.cfx* model in POSTFEKO.
6. Create a Cartesian graph and add the **S11** trace from the **SParamOpt** request.

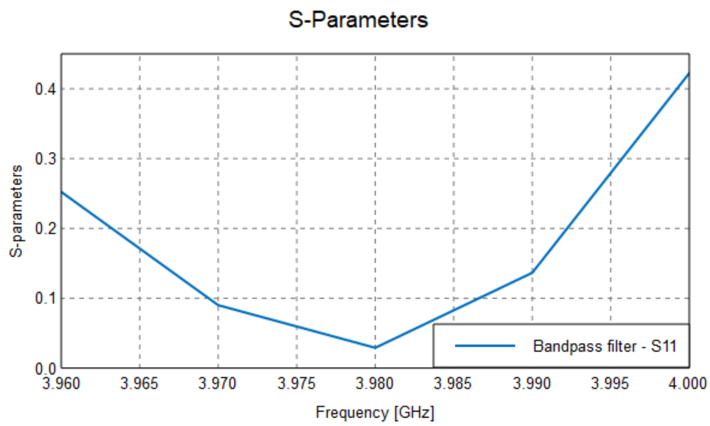
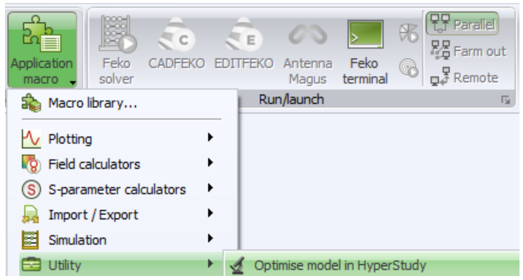


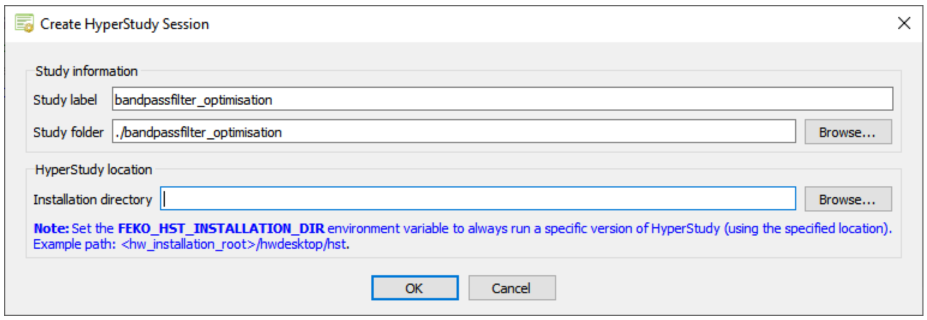
Figure 156: The reflection coefficient (S11) of the bandpass filter before optimisation from 3.960 GHz to 4 GHz.

7. Save the `bandpassfilter.pfs` file in the same location as the `bandpassfilter.cfx`.
8. Click **Application macro** in the **Scripting** group.
   The scripts loaded in the **Macro library** is displayed.

   *Figure 157: Example of the scripts available **Application macro** library.*



9. Click **Utility** and select **Optimise model in HyperStudy**.
The following **Create HyperStudy Session** dialog is shown.



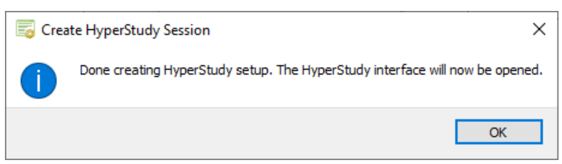
10. In the **Study label** field, enter a value for the name of the study.

11. In the **Study folder** field, specify a value for the directory of the HyperStudy session.

12. In the **Installation directory** field, specify the directory where HyperStudy is installed.

13. [Optional] Set the **FEKO_HST_INSTALLATION_DIR** environment variable to use a specific version of HyperStudy.

14. Click **OK** to start to create a HyperStudy session.
The following **Create HyperStudy Session** dialog is shown.



The extraction script `bandpassfilter.cfx_extract.lua` is created in the same directory as the `bandpassfilter.pfs`.

> **Note:** The trace on the Cartesian graph is extracted to the HyperStudy output file automatically. No additional scripting is required in the `bandpassfilter.cfx_extract.lua` file.

15. Click **OK** to launch HyperStudy.

16. Under **Define models**, verify the **Solver Execution Script** field is set to use the correct version.

> **Note:**
> - The script is accessible under **Edit > Register Solver Script**, which offers the possibility to register another solver or version.
> - The argument *-np* can be typed in **Solver Input Arguments** to specify the number of cores to use.

17. Under **Define Input Variables**, select which variables to include in the study. Only S1 – S3 should be activated and the default ranges used.
An example of the selected variables is displayed.

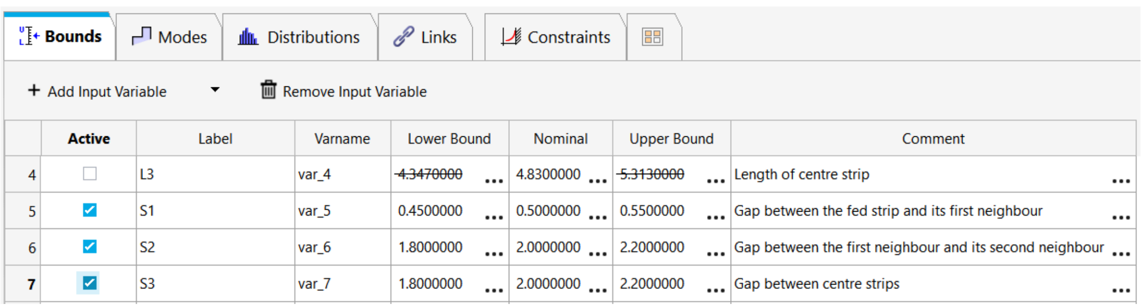


Figure 158: Example of the variable selection

18. Click **Next**.
19. Click **Run Definition**.

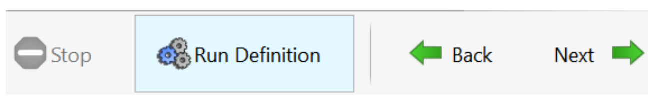


*Figure 159: Example selecting the **Run Definition** button.*

During the *Write* phase, the `bandpassfilter.cfx_extract.lua` file was copied to the run directory and executed after the Feko solver was run in the *Extract* phase. This generated an output file that HyperStudy can process easily.

> **Note:** The script `bandpassfilter.cfx_extract.lua` is different if a `.pfs` file was present before importing the variables and will automatically extract the visible traces on a Cartesian graph and polar graph.

See Figure 160 for the completed definition run for the **Write**, **Execute** and **Extract** phases for the initial test run done by HyperStudy.

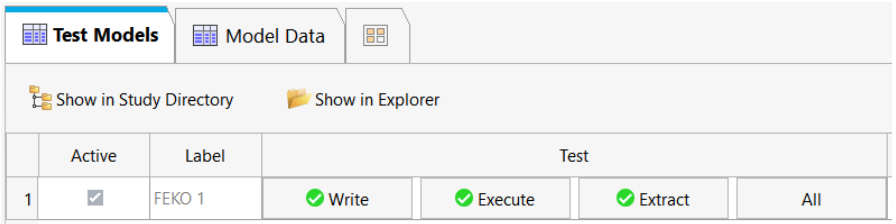
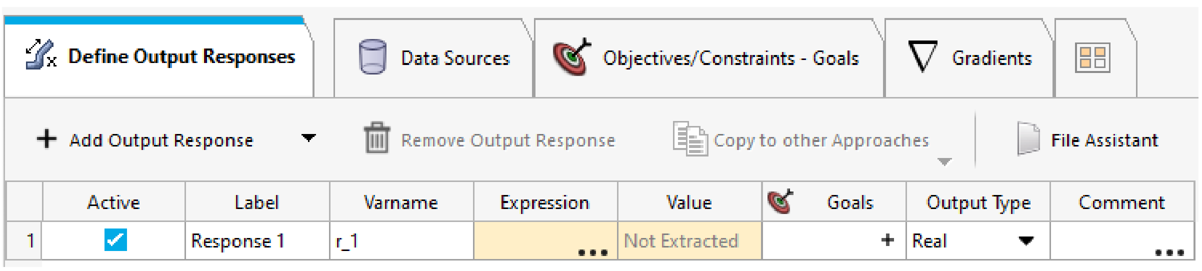


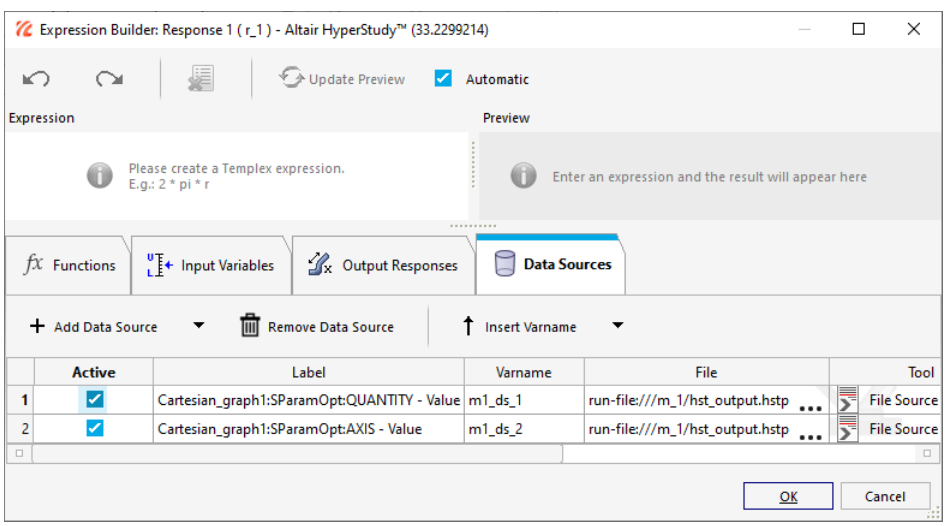
Figure 160: Example of the completed definition run.

20. Click **Next**.

21. Select **Add Output Response**.
An output response is added with the **Expression** field highlighted.



22. Select the **Expression** field.
The **Expression Builder:Response1(r1)** dialog is shown.



Note: The **Data Sources** *m1_ds_1* and *m1_ds2* is added from the POSTFEKO graph.

23. In the **Expression** field, enter *max(m1_ds_1)* and click **OK**.

24. Under **Goals** click + to add an optimisation goal.
The following dialog is displayed.

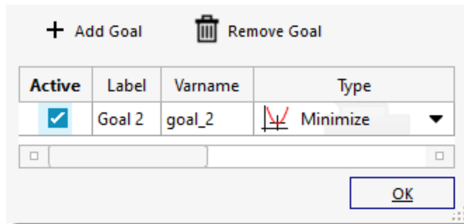


Figure 161: Example adding an optimisation goal.

25. Set the goal **Type** to **Minimize** and click **OK**.

26. Click **Evaluate** to extract the value from the output file.

*Figure 162: Example selecting the **Evaluate** button.*

HyperStudy is now configured to understand which model to use, which variables are available for modification and how to process the output.

27. Right-click on the defined study in the **Explorer** tab and click **Add**.

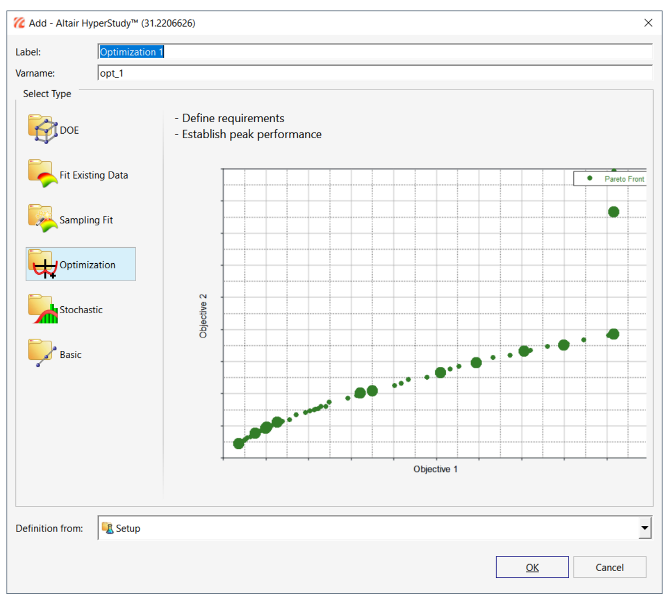


*Figure 163: The **Add** dialog and selecting an optimisation approach.*

28. Under **Select Type**, select **Optimization** and click **OK**.

29. In the **Definition from:** drop-down list, select **Setup** and click **OK**
The optimisation approach is created in the **Explorer** tab

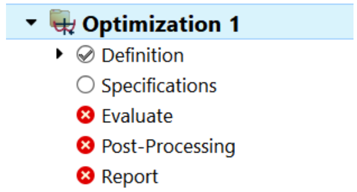
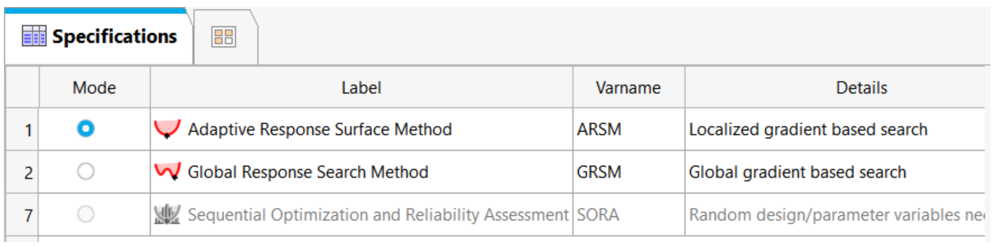


*Figure 164: Example of the **Optimisation 1** approach created.*

30. Click **Optimization 1** > **Specifications**, and select the **Adaptive Response Surface Method** as the optimiser.



	Mode	Label	Varname	Details
1		Adaptive Response Surface Method	ARSM	Localized gradient based search
2		Global Response Search Method	GRSM	Global gradient based search
7		Sequential Optimization and Reliability Assessment	SORA	Random design/parameter variables ne

*Figure 165: Selecting the **Adaptive Response Surface Method**.*

31. Click **Apply** and click **Next**.

*Figure 166: Example selecting the **Apply** button.*

32. Click **Evaluate Tasks**.

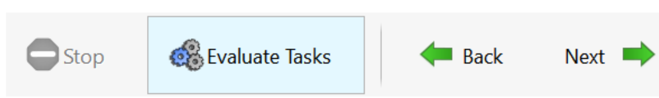


*Figure 167: Example selecting the **Evaluate Tasks** button.*

Each of the input variables is altered randomly, and its effect on the response analysed.

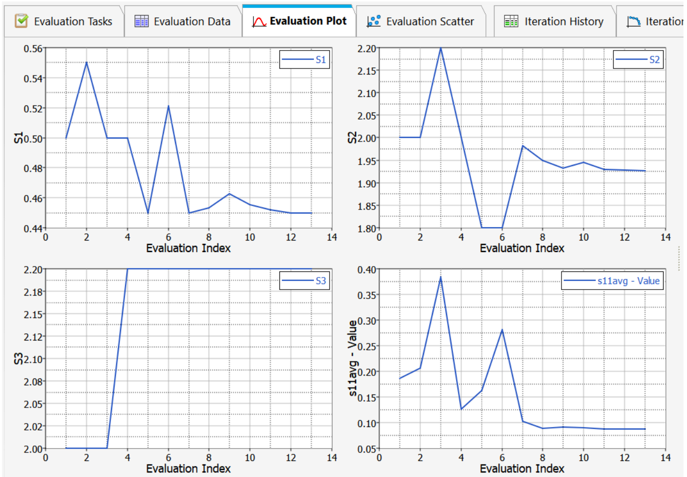


Figure 168: Typical progress data for an ARSM optimisation.

33. Click **Iteration History** tab and look for the row highlighted in green.
The optimum values are as follows:

- S1 = 0.4500000
- S2 = 1.9254786
- S3 = 2.2000000

I.3.4 Feko Modelling and Performance

Generate a model with the optimum values in Feko and solve.

The optimisation relied on five samples within the frequency range of interest, namely 3.96 - 4.00 GHz. To get a better sense of the frequency response for the optimum model, it is run in Feko.

1. Make a copy of the original CADFEKO model.
2. Update the following variables.
   - *S1* = `0.45`
   - *S2* = `1.9254786`
   - *S3* = `2.20`
3. Include the S-parameter configuration labelled **SParamBand** that is defined over the frequency range.
4. Mesh the model and run the Solver.

   The results show a reflection coefficient of less than -10 dB over the design frequency range.

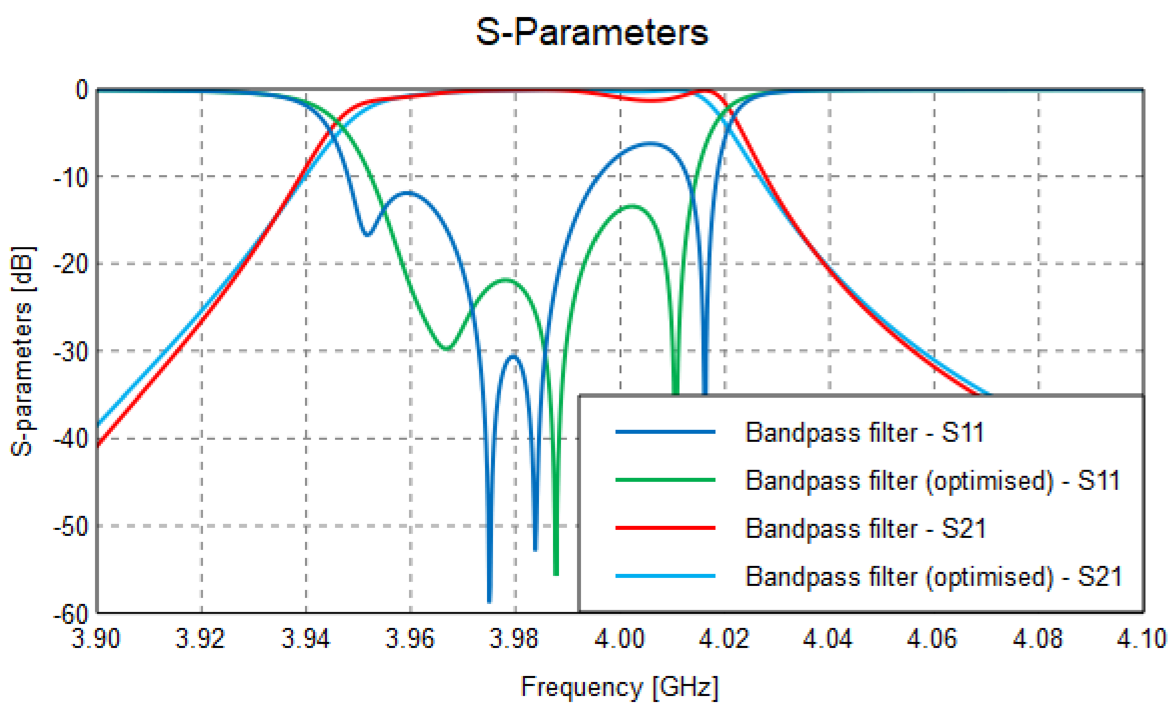


Figure 169: S-parameters for the bandpass filter between 3.9 – 4.1 GHz.

Index

Special Characters

A

B

C

S

T

U

V

W

Y