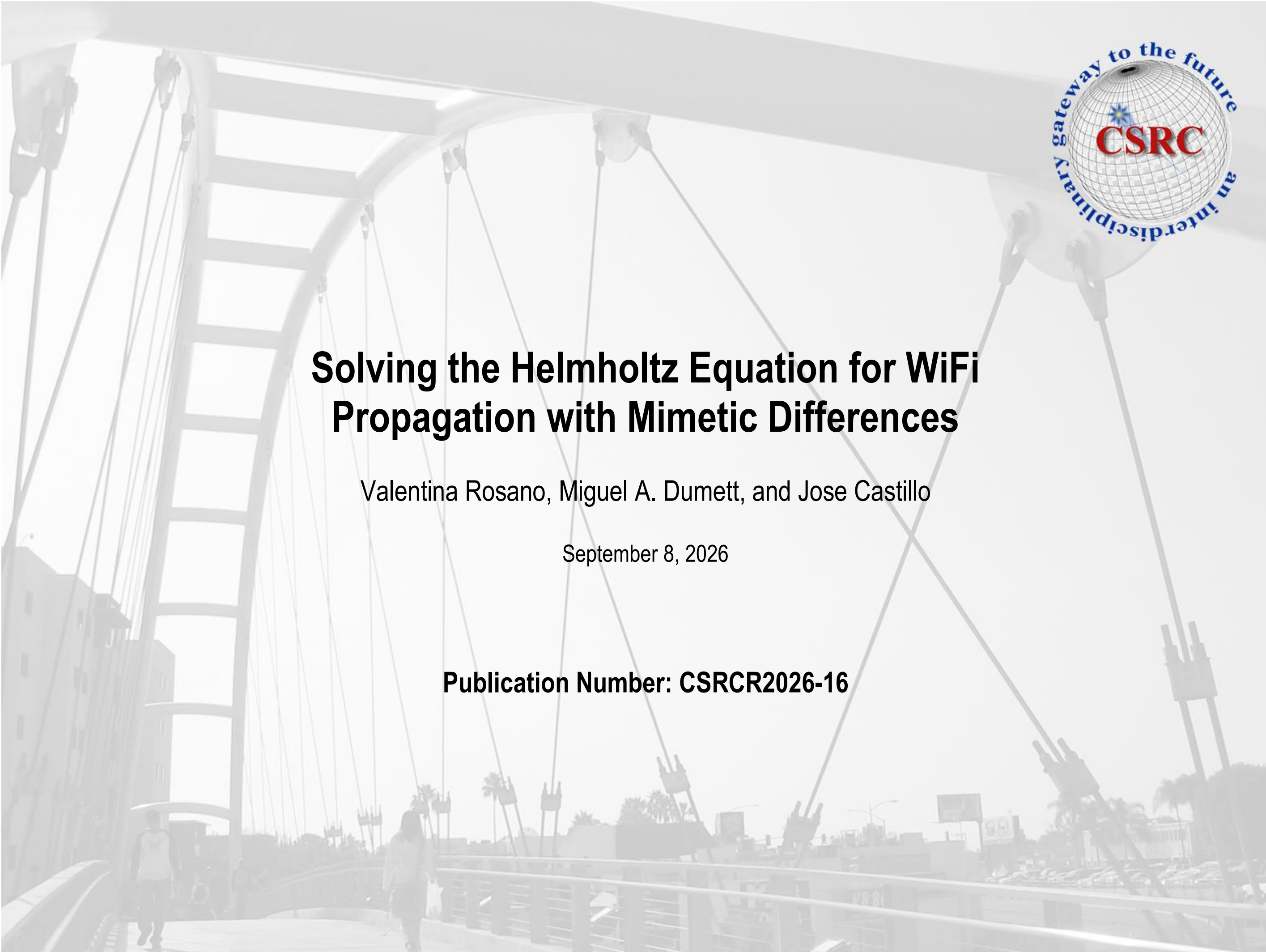


Solving the Helmholtz Equation for WiFi Propagation with Mimetic Differences

Valentina Rosano, Miguel A. Dumett, and Jose Castillo

September 8, 2026

Publication Number: CSRCR2026-16

Computational Science & Engineering Faculty and Students Research Articles

Database Powered by the Computational Science Research Center Computing Group & Visualization Lab

COMPUTATIONAL SCIENCE & ENGINEERING

SAN DIEGO STATE UNIVERSITY

Computational Science Research Center
College of Sciences
5500 Campanile Drive
San Diego, CA 92182-1245
(619) 594-3430

Solving the Helmholtz Equation for WiFi Propagation with Mimetic Differences

Valentina Rosano* Miguel A. Dumett † and José Castillo ‡

September 8, 2026

Abstract

In this paper we solve the homogeneous Helmholtz equation using mimetic differences. This equation models the propagation of the WiFi signal power.

1 Introduction

In this report, we consider the modeling of WiFi propagation using the Helmholtz equation in a domain with a heterogeneous refractive index. The Helmholtz equation is a well-known tool for modeling various complex physical phenomena involving wave propagation, with applications ranging from radar and sonar detection to medical and seismic imaging. Numerical simulations of wave phenomena are of particular interest due to their requirement for advanced numerical methods, especially when dealing with heterogeneous propagation domains. High-order methods have gained popularity in discretizing wave problems, as they help mitigate the pollution effect, allowing for consideration of high frequencies[4, 5, 6].

We discretize the Helmholtz equation using the 2D mimetic differences scheme proposed by Corbino and Castillo in [1]. In this scheme, the mimetic counterparts of the gradient and Laplacian differential operators are established on a staggered grid. Unlike conventional approaches that involve manual matrix creation or loops in finite difference methods, the mimetic method constructs these operators directly by specifying the grid and mimetic operators. As a result, solving partial differential equations becomes easier and straightforward.

2 Helmholtz equation and BC

We utilize the MOLE library [2] to implement a model of WiFi propagation in a two-dimensional flat. This example was taken from [3]. The flat model employed in the experiment features two walls:

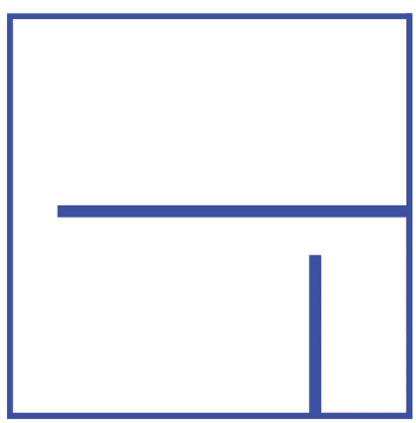


Figure 1: Flat model

The main point of this study is the impact of walls on the signal power.

*Universidad Nacional de Rosario, Argentina (valentina.rosano12@gmail.com).
†Computational Science Research Center at San Diego State University (mdumett@sdsu.edu)
‡Computational Science Research Center at San Diego State University (jcastillo@sdsu.edu).

WiFi signals are electromagnetic waves carrying internet data, typically described by Maxwell's equations. However, we employ a simplified model for our analysis. Assuming the time-dependence of an electromagnetic wave follows the form $\sin(\omega t)$ with the dispersion relation given by $\omega = k/n(x,y)$ for a refractive index distribution $n(x)$ and a constant angular wave number k, the electric field E satisfies the Helmholtz equation:

$$\Delta E + \frac{k^2}{n^2}E = 0 \qquad \text{in } D.$$

where $D = \Omega \setminus R$ represents the domain. Here $\Omega = [0, 40]^2$ denotes a square flat surface with a hole R where the hotspot is located (a circle of radius 1). Solving the equation inside the hotspot is excluded due to lack of physical interpretation.

To ensure a well-posed problem, the following boundary conditions are considered:

$$E = 1 \quad \text{on } \partial R, \qquad \frac{\partial E}{\partial n} = 0 \quad \text{on } \partial\Omega$$

These conditions assume that the only radiation source is the hotspot, as imposed by the boundary condition. Since there are no additional sources, the right-hand side of the Helmholtz equation vanishes.

3 Implementation

3.1 Refractive index

The refractive index $n(x,y)$ is 1 in the air, while it is a complex number in the walls whose real and imaginary parts have a physical interpretation:

- the real part wr defines the reflection on the wall (the amount of signal that does not pass),
- the imaginary part wa defines absorption of the wall (the amount that disappears).

Since the refraction coefficient depends on the point of the domain we're considering, we need our code to make this differentiation. In order to do this, we introduce a function $wall(X,Y)$ which returns 1 if the point (X,Y) is on the walls and 0 if it is on the air:

```
% define the walls
wall = @(X,Y) (X>=10.0 & X<=39.0 & Y>=20.0 & Y<=21.0) | ...
    (X>=30.0 & X<=31.0 & Y>=1.0 & Y<=16.0) | ...
    (X<=0.5 | X>=39.5 | Y<=0.5 | Y>=39.5);
```

Then the refraction index is given by

$$n(x,y) = 1 + (wr + wa \cdot i)\mathrm{wall}(x,y)$$

For our example we take

```
% wall reflection coefficient
wr = 0.7
% wall absorption coefficient
wa = 0.9*0.5
% angular wave number
wn = 6;
```

3.2 Domain

As we mentioned earlier, the domain D we're considering for our problem is a square Ω with a hole R in it, making it non-simply connected. Due to this characteristic and in order to simplify the implementation we use the MOLE library on the square Ω and therefore we've devised the following approach to address this.

Then, in first place, we define a grid on the square Ω.

```
%hotspot position
hsx = 2.0
hsy = 10.0
hsr = 1.0
```

```
% Spatial discretization
k = 2;   % Order of accuracy
m = 500;  % Number of cells along the x-axis
n = 500;   % Number of cells along the y-axis
a = 0;   % West
b = 40;   % East
c = 0;   % South
d = 40;   % North
dx = (b-a)/m;  % Step length along the x-axis
dy = (d-c)/n;  % Step length along the y-axis

% 2D Staggered grid
xgrid = [a a+dx/2 : dx : b-dx/2 b];
ygrid = [c c+dy/2 : dy : d-dy/2 d];
[X, Y] = meshgrid(xgrid, ygrid);
```

Now we are in position to introduce the coefficient $c = \frac{k^2}{n^2}$ (where k represents the wave number `wn`) of the 0-term of the equation. Taking into account the definition of n provided in the previous subsection, we have

```
% complex coefficient c=k^2/n^2
c = (wn^2./(1+(wr+i*wa)*wall(X,Y)).^2)';
c = reshape(c, (m+2)*(n+2), 1);
```

The nodes on the grid are numbered from the lower left corner to the upper right corner, row-wise from bottom to top and left to right within each row.

We define a vector called `freenodes`, which contains the indices of all nodes external to the hotspot.

```
% to detect the hotspot's position
HS = ((X-hsx).^2 + (Y-hsy).^2 < hsr^2)';
HS = reshape(HS, (m+2)*(n+2), 1);
ind = find(HS>0);
freenodes = setdiff(1:(m+2)*(n+2),ind);
```

Our objective now is to find an approximation of the solution E (which we have labeled `SOL`) in the nodes corresponding to the vector `freenodes`, while setting the value to 1 in the remaining nodes. Next, we compute the mimetic operator using the mimetic Laplacian, implemented in the MOLE library. Additionally, we apply the Neumann condition at the outer border of Ω.

```
% mimetic operator (Laplacian)
L = lap2D(k, m, dx, n, dy);
id_op = diag(sparse(c));
L = L + id_op;
L = L + robinBC2D(k, m, dx, n, dy, 0, 1); % Neumann BC
```

We solve the system for the nodes in `freenodes`, first setting the value to 1 in the nodesinside the hotspot.

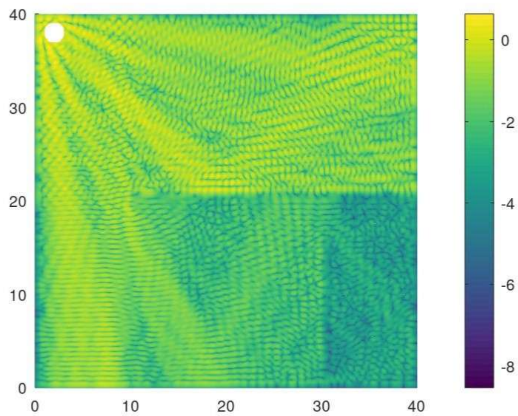
```
% RHS
RHS = zeros(m+2,n+2);
RHS = reshape(RHS, (m+2)*(n+2), 1);
RHS = RHS - L*HS;

SOL = ones((m+2)*(n+2),1);
SOL(freenodes) = L(freenodes,freenodes)\RHS(freenodes);
```

Finally, we plot the logarithm of the modulus of the solution, since it is complex. To highlight the position of the hotspot in the graphs, it is represented with a blank space.

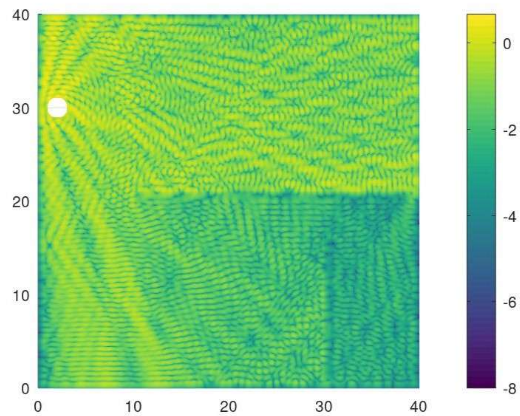
```
%graph the logarithm of the modulus of SOL
surf(X, Y, reshape(log(abs(SOL)), m+2, n+2)');
shading interp
colorbar
view(2)
```

4 Results

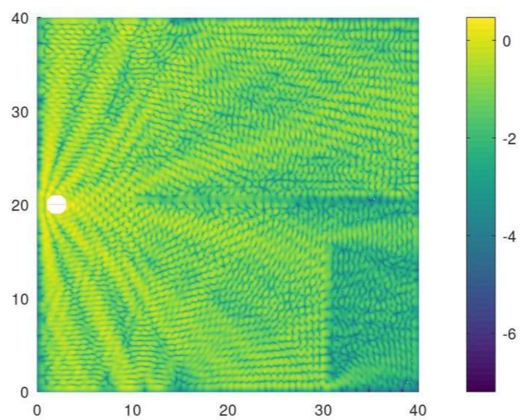
Moving the hotspot's position vertically along the $x = 2$ axis we obtain different distributions. We can observe six of these in the following graphs. By merely observing these images we can determmine the best place to install the hotspot is the center, aligned with the horizontal wall.



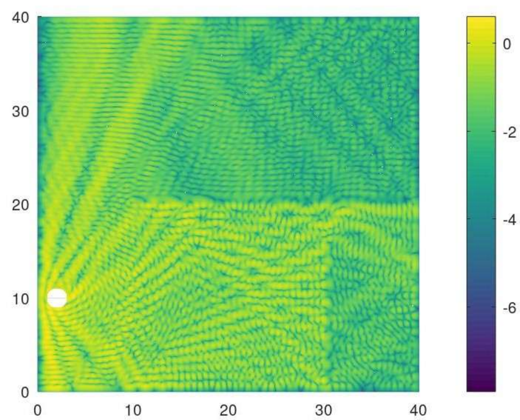
(a) hsy=38



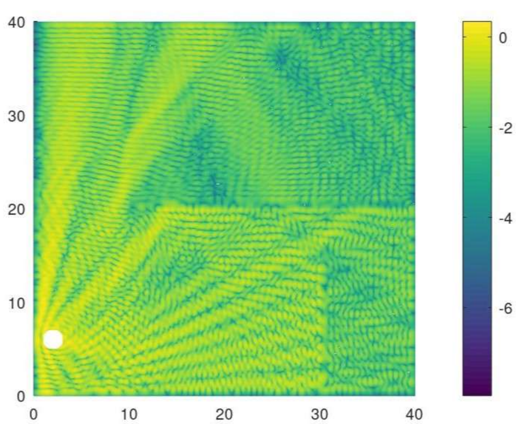
(b) hsy=30



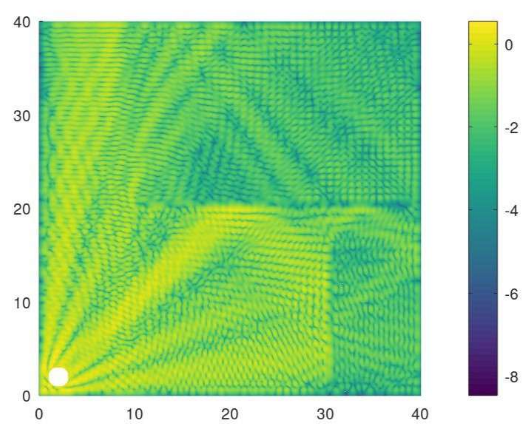
(a) hsy=20



(b) hsy=10



(a) hsy=6



(b) hsy=2

5 Conclusion

In general, mimetic methods performed well in approximating the solution of the Helmholtz equation, yielding coherent results comparable to those obtained using finite element methods in [3]. This suggests that mimetic methods can be a valuable tool for modeling and analyzing WiFi signal propagation in residential environments. Moreover, it is noteworthy that the results obtained are physically correct, aligning with our intuition regarding how WiFi signals would behave in such environments. However, we encountered a limitation in working with a non-connected domain.

For future work, our focus will be on comparing the numerical error between the mimetic and finite element methods. This comparative analysis will provide valuable insight into the strengths and limitations of each approach in accurately modeling WiFi propagation. Additionally, we aim to address the issue of non-connectedness in the domain, which poses a challenge in our current modeling framework.

6 Acknowledgment

This work was done at the University of Rosario, Argentina during Professor Jose Castillo Fulbright Visit in 2023.

References

[1] Corbino, J., and Castillo, J.E., High-order mimetic finite-difference operators satisfying the extended Gauss divergence theorem, Journal of Computational and Applied Mathematics 364 (2020). 112326

[2] Corbino, J., and Castillo, J.E., MOLE: Mimetic Operators Library Enhanced: The Open-Source Library for Solving Partial Differential Equations using Mimetic Methods, 2017. GitHub, `https://github.com/csrc-sdsu/mole`.

[3] F. Hetch, FreeFEM Documentation, Release 4.12.

[4] F. Ihlenburg and I. Babuska, Finite element solution of the Helmholtz equation with high wave number. II. The h-p version of the FEM, SIAM J. Numer. Anal. 34 (1997), no. 1, 315–358, DOI 10.1137/S0036142994272337. MR1445739

[5] J. M. Melenk and S. Sauter, Convergence analysis for finite element discretizations of the Helmholtz equation with Dirichlet-to-Neumann boundary conditions, Math. Comp. 79 (2010), no. 272, 1871–1914, DOI 10.1090/S0025-5718-10-02362-8. MR2684350

[6] J. M. Melenk and S. Sauter, Wavenumber explicit convergence analysis for Galerkin discretizations of the Helmholtz equation, SIAM J. Numer. Anal. 49 (2011), no. 3, 1210–1243, DOI 10.1137/090776202. MR2812565