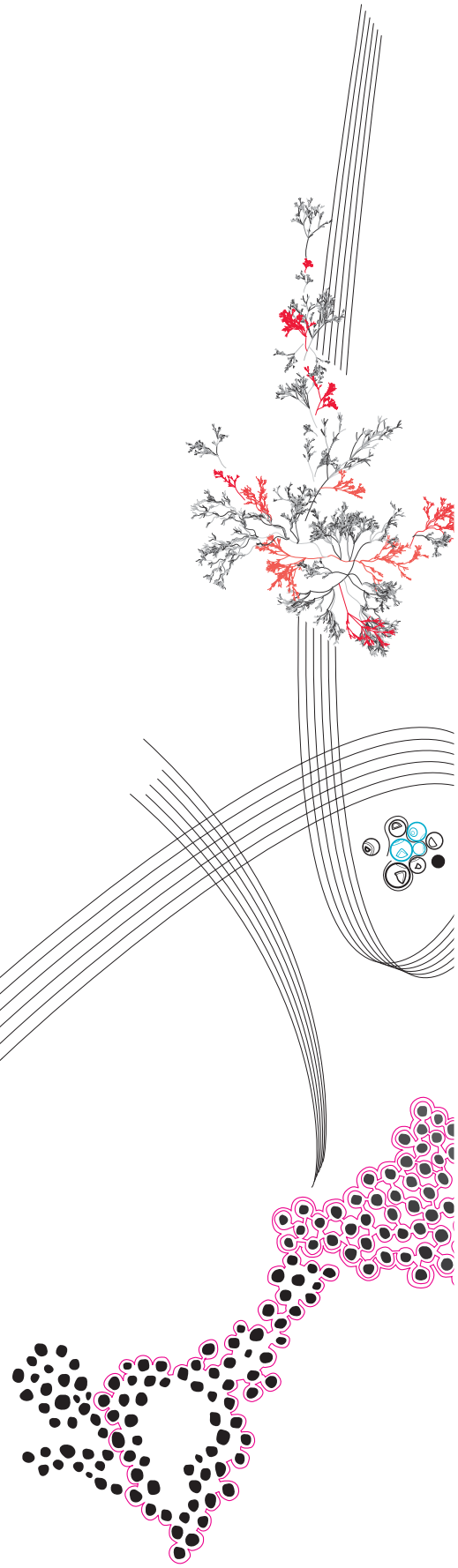

BSc Thesis Applied Mathematics

Performance comparison of FAME and MPB for simulating wave bands of photonic crystals

Serge Iwan Johanns

Supervisor: M. Kozon, K. Chen

February, 2022

Department of Applied Mathematics and
Department of Technical Computer Science
Faculty of Electrical Engineering,
Mathematics and Computer Science

UNIVERSITY OF TWENTE.

Preface

This article was written as my bachelor's thesis in Technical Computer Science and Applied Mathematics.

I want to thank Marek Kozon and dr. ir. Kuan-Hsun Chen for their continued support throughout this process, and especially for their invaluable help in evaluating and adjusting my process, planning and writing. Additionally I want to thank Marek for his help with MPB, for providing the MPB scripts, for generating the reference data with which the results of FAME and MPB were compared, and for generally being incredibly helpful with all of the issues that cropped up during the project.

I also want to thank the University of Twente's Faculty of Electrical Engineering, Mathematics and Computer Science for making their Epyc supercluster available for me to run the simulations on, and Frederik Reenders from technical support for helping me with the many problems I encountered.

Finally, I want to preemptively thank dr. Jan-Kees van Ommeren for taking the time to read this thesis, even though at the time of writing he hasn't done that yet.

Performance comparison of FAME and MPB for simulating wave bands of photonic crystals

Serge I. Johanns

February, 2022

Abstract

In this article we compare the computational efficiency and accuracy of the Fast Algorithm for Maxwell's Equations and the MIT Photonic Bands package. We apply the packages to three unique crystals at different simulation resolutions and compare the results to high-resolution reference data. In general FAME achieves lower computation times per wave-vector than MPB, although the relative difference decreases with higher resolutions. FAME exhibits higher error for two out of three crystals regardless of resolution, and for one crystal FAME exhibits an MSE that is several orders of magnitude higher than that of MPB. From these results, FAME could be a situational replacement for MPB, but does not have a clear advantage in the general case. *Keywords*: photonic crystals, Maxwell's equations, MIT Photonic Bands

Contents

1 Introduction

Photonic crystals (PhCs) are materials with periodically changing refractive indices. PhCs have widely used industrial applications [1, 2], and while three-dimensional (3D) PhCs are not yet commercialized several industrial applications have already been proposed [3]. As such, PhCs are an active field of research, and there is much demand for simulation software for PhCs, including software that can simulate 3D PhCs.

The behavior of light in such 3D PhCs can be described with the 3D Maxwell's equations. The MIT Photonic-Bands (MPB) program has been one of the most widely used programs for simulating such photonic bands in 3D PhCs using Maxwell's equations for over 20 years [4, 5, 6, 7, 8]. In recent years, however, the authors of [9] have been working on a new Fast Algorithm for Maxwell's Equations (FAME) that has the potential to provide a significant performance increase for PhCs. If true, this would be valuable for research, since simulations using MPB can take impractical amounts of time for realistic crystal sizes.

However, because the FAME package is relatively new it may not be clear to optics researchers whether this new method is actually faster than MPB, and if so, to what extent. Furthermore, since the numerical methods used in FAME and MPB might differ in accuracy, it is unclear whether the accuracy of FAME is sufficient to be a substitute for MPB.

In this paper, we perform an analysis of the computational complexity of both FAME and MPB by applying the packages to three different crystal structures. We also compare the results in order to determine if FAME has a significant difference in accuracy compared to MPB. This information will allow researchers to make an informed decision on whether to use FAME or MPB for computing the behavior of light in PhCs.

1.1 Physics of photonic crystals

A photonic crystal lattice is a Bravais lattice, meaning the lattice is made of a primitive cell Ω_p that is repeated and translated by integer multiples of the primitive translation vectors. This if the translation vectors are $\tilde{\mathbf{a}}_1$, $\tilde{\mathbf{a}}_2$, and $\tilde{\mathbf{a}}_3$, then the primitive cell is repeated at every $k\tilde{\mathbf{a}}_1 + l\tilde{\mathbf{a}}_2 + m\tilde{\mathbf{a}}_3$ where $k, l, m \in \mathbb{Z}$. This primitive cell is formed by the set $\Omega_p := \{a\tilde{\mathbf{a}}_1 + b\tilde{\mathbf{a}}_2 + c\tilde{\mathbf{a}}_3 : (a, b, c) \in [0, 1]^3\}$. The number of translation vectors is equal to the number of dimensions of the photonic crystal, so while a 3D crystal uses 3 lattice vectors, a 2D crystal can be described with only 2, *etc.* The crystal is defined the primitive cell's shape, size, and material distribution, where vacuum is one of the possible materials. These materials can have different refractive indices, which gives the crystal it's interesting properties.

We model the propagation of light through a photonic crystal using the 3D time-harmonic Maxwell's equations [9]:

$$\begin{aligned} \nabla \times E &= i\omega B, & \nabla \times H &= -i\omega D, \\ \nabla \cdot B &= 0, & \nabla \cdot D &= 0, \end{aligned} \tag{1}$$

where E is the electric field, H is the magnetic field, B is the magnetic density, D is the

flux density, and ω is the frequency. These satisfy the constitutive relations:

$$B = \mu H + \zeta E \text{ and } D = \varepsilon E + \xi H, \tag{2}$$

where ε is the electric permittivity and μ is the magnetic permeability. In this article we only consider materials that are isotropic, meaning ε is constant within a material, nondispersive, meaning waves of all frequencies travel at the same speeds in the material, and where $\mu = 1$ and $\zeta = \xi = 0$.

A PhCs photonic band structure is the function $\omega = \omega(k)$. It describes the frequencies that ω can attain depending on the direction the light travels in, called the wave-vector. The band structure is normally given along a path of wave-vectors called the Brillouin path, which is determined by the type of Bravais lattice the crystal has and crosses points of high symmetry.

2 Summary of the FAME package

In the following section we provide a cursory overview of the FAME algorithm as described in [9]. FAME computes band structures of a crystal given the lattice structure, which is characterized by the lattice translation vectors $\tilde{\mathbf{a}}_1$, $\tilde{\mathbf{a}}_2$, and $\tilde{\mathbf{a}}_3$, the electric permittivities in the material and in the vacuum, respectively ε_{in} and ε_o.

Broadly speaking FAME is based on using Yee's finite-difference scheme [10] to discretize the operators used in Maxwell's equations, such as the curl operator. With this scheme each cell is split into a grid with mesh sizes n_1, n_2, n_3 along the x, y, and z axes respectively. Then the partial derivative at a point is approximated by the difference of the next adjacent points. This discretized form leads to a Generalized Eigenvalue Problem (GEP) which, for the materials considered in this article, simplifies to

$$C^*C\mathbf{e} = \lambda B_\varepsilon \mathbf{e},$$

and

$$CB_\varepsilon^{-1}C^*\mathbf{h} = \lambda \mathbf{h}.$$

The FAME algorithm calculates the band structure using the eigenpairs obtained from solving this GEP. Each eigenvalue corresponds to a frequency in the band structure, specifically $\lambda = \omega^2$, and the eigenvector to the corresponding field, $\mathbf{e}$ being electrical and $\mathbf{h}$ magnetic.

Firstly, FAME addresses the issue that the lattice translation vectors do not necessarily align with the Cartesian coordinates, which means discretizing over the lattice cell Ω_p directly may induce a large number of stair errors [11]. To correct this, we derive an equivalent cuboid computational cell. Firstly, QR factorization with column pivoting is used to take

$$\tilde{Q}R = \begin{bmatrix}\tilde{\mathbf{a}}_1 & \tilde{\mathbf{a}}_2 & \tilde{\mathbf{a}}_3\end{bmatrix}\Pi,$$

where $\tilde{Q}$ is orthonormal, R is upper triangular, and Π is a permutation matrix. R corresponds to the vectors which have been rotated so that the first vector aligns with the Cartesian x-axis and the second vector is in the xy-plane, $\tilde{Q}$ is the rotation matrix which produces the necessary rotation, and Π is the permutation matrix which permutes the columns to match the new order after column pivoting. Then we let

$$S := \operatorname{diag}(\operatorname{sign}(R_{1,1}), \operatorname{sign}(R_{2,2}), \operatorname{sign}(R_{3,3})),$$

so that SR has a positive diagonal. This ensures the computational cell we work with is positive. Then we let $\begin{bmatrix}\mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3\end{bmatrix} = SR$. In order to make sure the rotation is still correct we let $Q := \tilde{Q}S$, which cancels out the adjustment of R. These new vectors represent x, y, and z respectively, but are not yet orthogonal. That is why we take the computational cell $\Omega_c = [0, \mathbf{a}_1(1)] \times [0, \mathbf{a}_2(2)] \times [0, \mathbf{a}_3(3)] \subset \mathbb{R}^3$ (where $\mathbf{a}_l(n)$ is the n-th component of $\mathbf{a}_l$), which is cuboid and therefore aligns properly with the Cartesian coordinates, mitigating error.

For the next step, we generate $\Omega_{p,m}$, the set of points in Ω_p that are actually inside of the nonvacuum material, instead of in the vacuum. This construction is defined in [12] as a union of spheres and cylinders of varying lengths and radii that make up the crystal structure. Finally, using this definition of $\Omega_{p,m}$, we derive a diagonal matrix E_ε that represents the electric permittivity at each sampling point for the finite-difference scheme. Since the finite-difference scheme splits every cell into an $n_1 \times n_2 \times n_3$ grid we map a grid point $\mathbf{r}_c \in \Omega_c$ onto a point in the primitive cell. We use

$$\tilde{\mathbf{c}} = \Pi \begin{bmatrix}\mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3\end{bmatrix}^{-1} \mathbf{r}_c,$$

and

$$\mathbf{r}_p = \begin{bmatrix}\tilde{\mathbf{a}}_1 & \tilde{\mathbf{a}}_2 & \tilde{\mathbf{a}}_3\end{bmatrix} (\tilde{\mathbf{c}} - \lfloor \tilde{\mathbf{c}} \rfloor).$$

We use $\tilde{\mathbf{c}} - \lfloor \tilde{\mathbf{c}} \rfloor$ instead of $\tilde{\mathbf{c}}$ to translate the point into the primitive cell (this is permitted due to periodicity of the crystal structure). Then the corresponding electric permittivity is ε_{in} if $\mathbf{r}_p \in \Omega_{p,m}$ and ε_o otherwise. Using this projection we can define B_ε as

$$B_\varepsilon = \begin{bmatrix} B_{c,x} & & \\ & B_{c,y} & \\ & & B_{c,z} \end{bmatrix},$$

where if $i = 0, \ldots, n_1 - 1$, $j = 0, \ldots, n_2 - 1$, $k = 0, \ldots, n_3 - 1$ represent the x, y, and z indices within a cell, $l = 1 + i + n_1 j + n_1 n_2 k$, and

$$\delta_x = \frac{\mathbf{a}_1(1)}{n_1}, \delta_y = \frac{\mathbf{a}_2(2)}{n_2}, \delta_z = \frac{\mathbf{a}_3(3)}{n_3},$$

then $B_{c,x}(l, l)$, the diagonal element on the l-th row and column of $B_{c,x}$, is the electric permittivity corresponding to the grid point $\mathbf{r}_c = [(i - \frac{1}{2}\delta_x, j\delta_y, k\delta_z]$, and similarly for $B_{c,y}$ and $B_{c,z}$, except that the $\frac{1}{2}$ offset is with respect to the corresponding axis. All non-diagonal entries are 0, and so B_ε is also diagonal. This offset and threefold repetition in B_ε is in accordance with Yee's finite-difference scheme, since the matrix will be used together with the discretization of the curl operator C.

Next, we perform a Singular Value Decomposition of C in order to significantly speed up matrix operations. C has the form

$$C = \begin{bmatrix} 0 & -C_3 & C_2 \\ C_3 & 0 & -C_1 \\ -C_2 & C_1 & 0 \end{bmatrix}.$$

The definitions of C_l are outside of the scope of this introduction[1], however a cursory explanation is as follows:

[1]For details see [13].

Let $\mathbf{E} = \begin{bmatrix} E_1(\mathbf{x}) & E_2(\mathbf{x}) & E_3(\mathbf{x}) \end{bmatrix}^T$ denote the function for the electric field. In order to represent the offset with respect to each dimension used in Yee's finite-difference scheme, we define $\mathbf{x}_l(i,j,k) = (i\delta_x, j\delta_y, k\delta_z) + \Delta_l$ where $\Delta_1 = (\frac{\delta_x}{2}, 0, 0), \Delta_2 = (0, \frac{\delta_y}{2}, 0), \Delta_3 = (0, 0, \frac{\delta_z}{2})$. Then each $E_l(\mathbf{x})$ is sampled at the points $\mathbf{x}_l(i,j,k)$, and the point $\mathbf{x} = (i\delta_x, j\delta_y, k\delta_z)$ corresponds to the index $m = i + jn_1 + kn_1n_2$, meaning $E_l(\mathbf{x}_l(:,:,:))$ is arranged in column-major order [13]. Using these we can discretize the partial derivatives of $\mathbf{E}$:

$$
\begin{aligned}
C_1 E_l(m) &= \frac{E_l(\mathbf{x}_l(i+1,j,k)) - E_l(\mathbf{x}_l(i,j,k))}{\delta_x}, && l = 2,3, \\
C_2 E_l(m) &= \frac{E_l(\mathbf{x}_l(i,j+1,k)) - E_l(\mathbf{x}_l(i,j,k))}{\delta_y}, && l = 1,3, \\
C_3 E_l(m) &= \frac{E_l(\mathbf{x}_l(i,j,k+1)) - E_l(\mathbf{x}_l(i,j,k))}{\delta_z}, && l = 1,2,
\end{aligned}
$$

thus the matrices C_l represent using Yee's finite difference scheme to approximate the partial derivative w.r.t. x, y, and z.

By finding eigenvectors of submatrices C_l we can construct a matrix T that diagonalizes all three C_l-s simultaneously so that $C_l T = T\Lambda_l$. Finally we use these to derive U_r, Σ_r, V_r such that $C = U_r \Sigma_r V_r^*$. we can emulate multiplication of T with a vector using FFT-based matrix-vector multiplication, which allows for much faster multiplication and does not require loading all of T into memory, since that is infeasible. This multiplication will be necessary while solving the GEP.

The GEP that follows directly from Maxwell's equations has a rather large null-space, meaning the null space makes up for approximately one third of all eigenvalues. This makes it inefficient to solve directly, since the large null space affects the convergence to the desired solution of the GEP. However, we can deflate the null space by rewriting the GEP in terms of the matrices derived in the SVP, obtaining the null-space free standard eigenvalue problem (NFSEP) [14]

$$
A_r \mathbf{x} = \left(\Sigma_r V_r^* B_\varepsilon^{-1} V_r \Sigma_r \right) \mathbf{x} = \lambda \mathbf{x}
$$

and eliminating a third of the eigenvalues. This NFSEP can be solved using the inverse Lanczos method. This yields eigenpairs for the preconditioned problem, after which we can derive the correct electrical field with $\mathbf{e} = B_\varepsilon^{-1} V_r \Sigma_r \mathbf{x}$. This yields the target eigenpairs. From the eigenvalues we can calculate the band structure, commonly only taking the smallest few since these are often the most defining.

3 Methodology

In order to adequately compare FAME and MPB we will consider both the computation times for equivalent resolutions, ranging from $4 \times 4 \times 4$ to $128 \times 128 \times 128$, except in the 2D case, where the third dimension resolution is trivially 1 and the other resolutions are chosen more 'spaced-out' from 20 to 100, since time measurements at too low resolutions might be inaccurate. These metrics will be compared for three selected crystal structures, depicted here. Figure 1 shows the 2-dimensional crystal, Figure 2 shows the commonly used 'inverse woodpile' structure, and Figure 3 shows a regular woodpile structure with blocks instead of cylinders, to compare the accuracy when sharp edges are involved. Note that the latter two images show the approximate form of the crystal, but the crystals

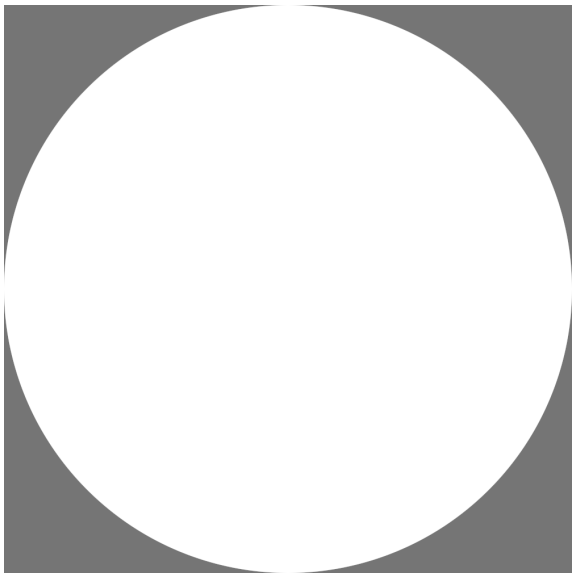

FIGURE 1: The 2D structure.

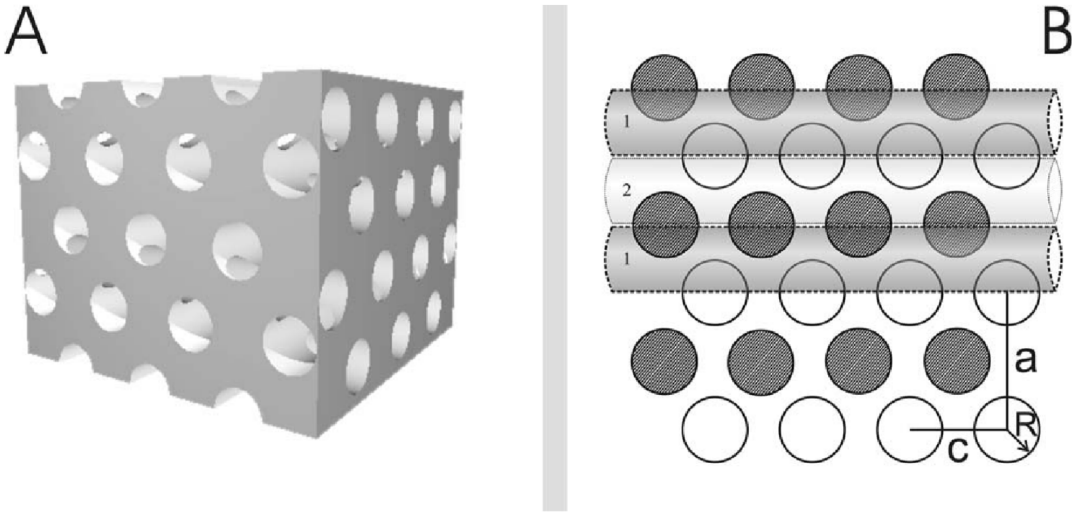

FIGURE 2: The inverse woodpile structure. Source: [15]

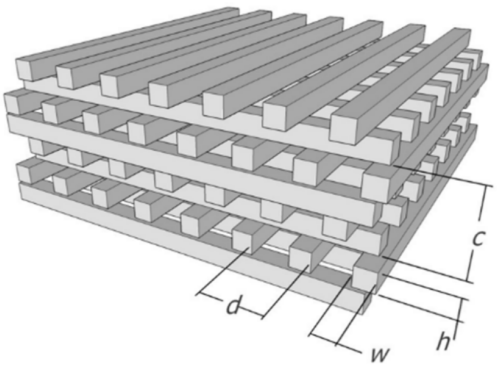

FIGURE 3: The inverse woodpile structure. Source: [16]

used in this article are subtly different. In the first part of this section we will list the exact properties of the three selected structures, in the second part we will discuss the methodology for comparing computing time, and in the second part we will cover accuracy.

The 2D crystal has a square lattice of 1×1 and is filled with a material with an electric permittivity of 12.1, except in the region within the inscribed circle centered at $\left(\frac{1}{2}, \frac{1}{2}\right)$ with radius $\frac{1}{2}$, where the permittivity is 1. The inverse woodpile has a cubic lattice of $1 \times 1 \times 1$, filled with material with a permittivity of 12.1 except in the regions contained in the four cylinders: two centered along the lines $\left(t, \frac{1}{2}, 0\right)$ and $\left(t, 0, \frac{1}{2}\right)$, respectively, and two centered along the lines $\left(\frac{1}{2}, \frac{1}{4}, t\right)$ and $\left(0, \frac{3}{4}, t\right)$, all with radius 0.24. Every point contained in one or more of these cylinders has an electric permittivity of 1. Finally, the sharp woodpile is essentially the same except the cylinders are blocks that are centered on the same lines but have a total width and height of $\frac{1}{4}$, ensuring they touch but do not overlap. It is also not inverse, so the material inside the blocks has a permittivity of 10.89 while the material outside the blocks has a permittivity of 1.

Next we compare the computing time required for the selected crystals. These are simulated at several resolutions and the average execution time per wave-vector is measured (only the time spent on individual wave-vectors, not initialization time). All simulations are performed on the same supercluster with two AMD Epyc™ 7742s and two NVIDIA Quadro RTX™ 6000s to ensure comparable results. However, FAME-GPU is the only implementation capable of leveraging GPU acceleration, which gives an inherent advantage over MPB. Since this is a constant factor difference the measurements still meaningfully indicate the computational complexity of the algorithm. Furthermore, the model of GPU is quite common in high-end scientific computing, which means the computing time required for FAME-GPU with GPU acceleration is critical for determining whether FAME is an adequate replacement for MPB for optical researchers. The computing times are retrieved from the outputs of the respective software packages. FAME outputs the time required for each wave-vector, which are added to calculate the average. With MPB the initialization time is subtracted from the total time and the result is divided by the total amount of wave-vectors to determine the average. This is because MPB only gives time in whole seconds, so results for individual wave-vectors at small resolutions are not meaningful.

Finally we compare the accuracy of the resulting band structures. We compare both the frequencies that make up the band structures, which are equivalent to the eigenvalues the NFSEP, as well as the electrical fields themselves. In order to determine accuracy we compare these against the values generated by a reference simulation using MPB, at a resolution of $240 \times 360 \times 240$ for the 3D crystals and $1000 \times 1000 \times 1$ for the 2D case, both much higher than the testing simulations. This ensures the error induced by numerical approximations is much lower, and thus the error between the reference value and the true value will not significantly affect the magnitude of the error between the reference values and the testing values. The scripts for the reference values were provided by M. Kozon, and used identically for the MPB tests to ensure the structures were still the same.

The eigenvalues are real numbers, and for each wave-vector there is one for each band, so an adequate way to compare them is by taking the Mean Squared Error (MSE) of all values. For a given resolution, if B is the set of bands, W is the set of wave-vectors, and $\lambda_{test}(w, b)$ and $\lambda_{ref}(w, b)$ are the eigenvalues at wave-vector w corresponding to band b in

the test measurements and reference data respectively, then the MSE is given by:

$$\sigma_m := \frac{1}{|W|} \sum_{w \in W} \frac{1}{|B|} \sum_{b \in B} (\lambda_{test}(w,b) - \lambda_{ref}(w,b))^2. \tag{3}$$

With this we can compare the MSEs of FAME and MPB for all tested resolutions. In order to compare individual frequency bands we will also use relative error:

$$\sigma_r(w,b) := \frac{|\lambda_{test}(w,b) - \lambda_{ref}(w,b)|}{\lambda_{ref}(w,b)}. \tag{4}$$

This will allow us to see if a pattern in the MSE is caused by a trend among all frequency bands or by an outlier with a high error. Of course (4) is undefined if $\lambda_{ref}(w,b) = 0$, but in practice this is not an issue since there are very few w and b where $\lambda_{ref}(w,b) = 0$, and in those cases both FAME and MPB also produce a $\lambda_{test}(w,b)$ of 0, regardless of resolution. Since there are far too many frequency bands to plot at once for both FAME and MPB we will only consider the lowest and highest calculated frequency band, the highest calculated being the 8th lowest of all of the possible bands.

4 Results

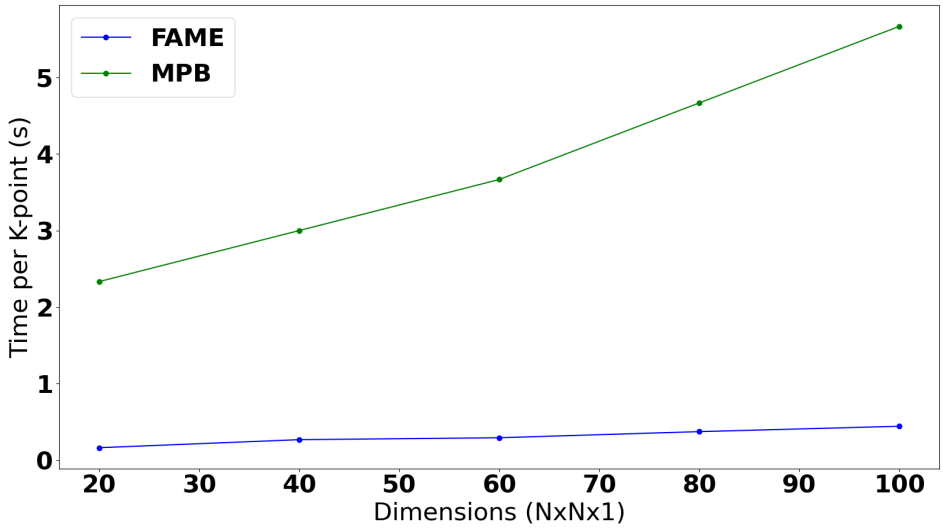

FIGURE 4: Required time for the 2D crystal.

In Figure 4 we see that. FAME is not in the same order of magnitude as MPB for any of the resolutions. MPB also increases significantly, but not necessarily linearly with the number of cells, considering that the time required doubles while the number of cells increases from $20 \times 20 = 400$ to $100 \times 100 = 10000$.

In Figure 5 we see that with the addition of a third dimension we see the required time for FAME increases much faster, just like the required time for MPB. This is likely because the total number of cells increases much faster now that there are three dimensions, and so the total number of calculations is orders of magnitude higher. It seems there is some

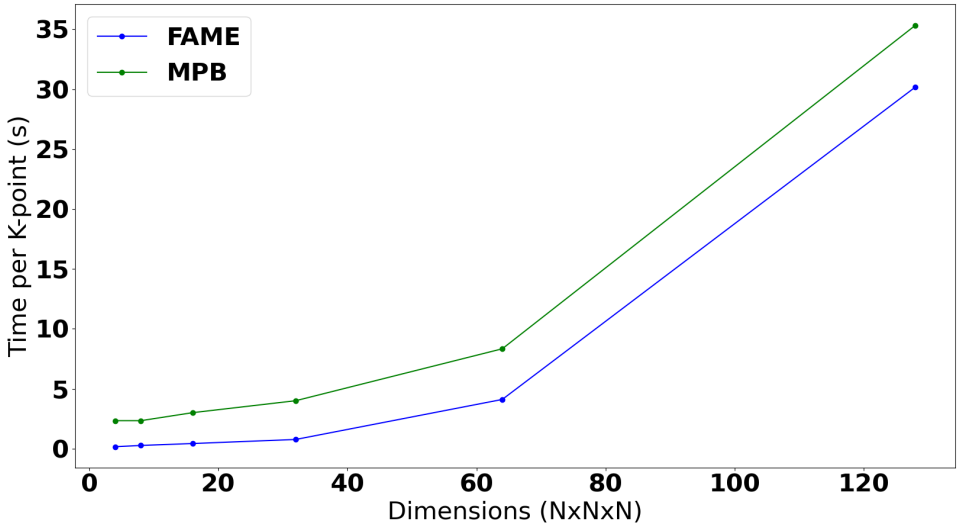

FIGURE 5: Required time for the inverse woodpile.

constant time per wave-vector that MPB requires and FAME does not. This could be because MPB's initialization per wave-vector is much more involved, or it could be one of the sources of error outlined in Section 5.

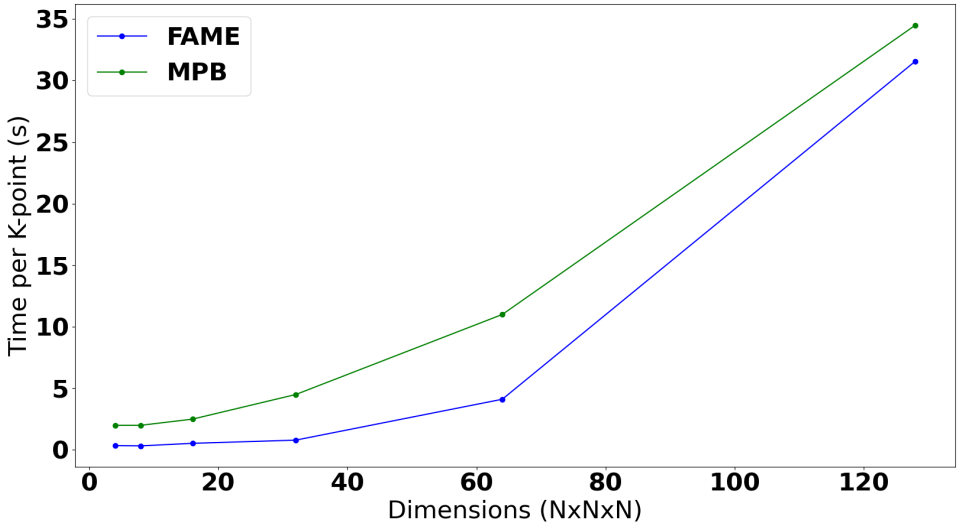

FIGURE 6: Required time for the sharp woodpile.

In Figure 6 the results are very comparable to Figure 5. This is to be expected, since both have the same number of cells.

In general it seems that FAME performs faster by an approximately constant amount, which means the advantage of using fame is reduced more the higher the resolution. Regardless, FAME always seems to perform slightly better, and especially at lower reso-

lutions, which could be especially valuable when running many different lower resolution simulations, for instance in an automated search, for instance in an automated search.

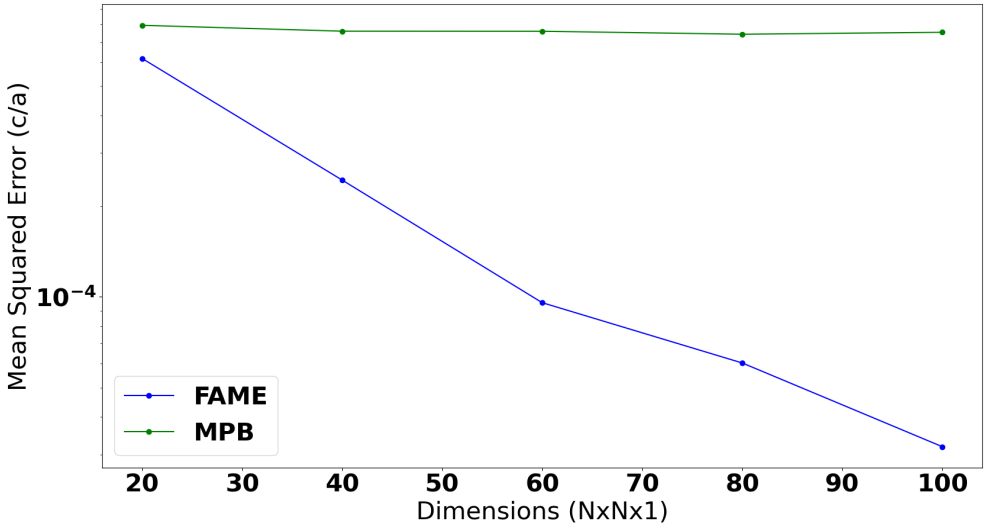

FIGURE 7: MSE of the eigenvalues for the 2D crystal.

In Figure 7 the error of MPB barely changes. This could be because the crystal is 2D the total number of cells increases less quickly. Thus, the error values for the 2D case do not change as quickly as for the 3D cases. This is corroborated by the fact that the error of FAME also ranges much less than in the 3D cases. This is also the only simulation where FAME's error is lower than MPB's, but given the small spread that could simply be coincidental.

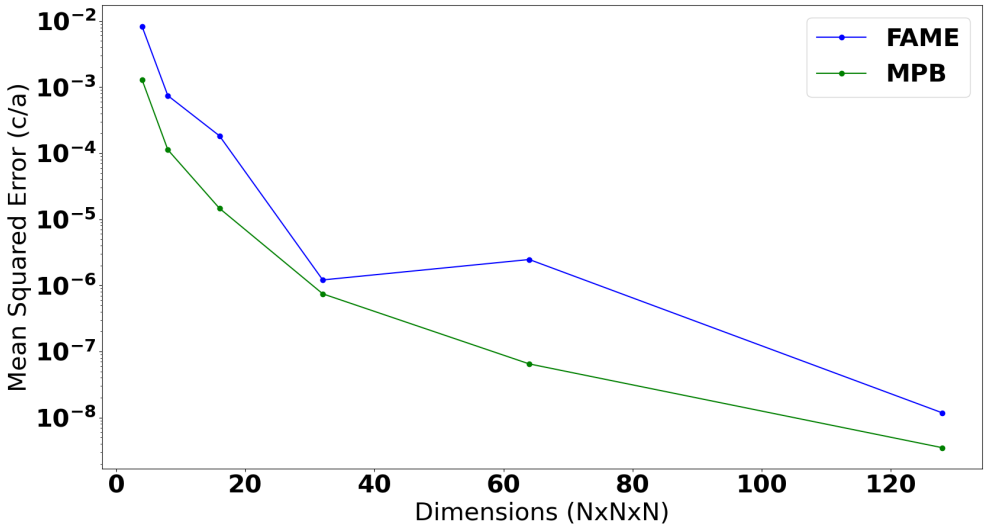

FIGURE 8: MSE of the eigenvalues for the inverse woodpile.

In Figure 8 we see FAME systematically has a higher error than MPB, but both decrease at approximately the same rate and are usually within an order of magnitude. Thus it

seems both FAME and MPB converge to the same reference value but MPB does so faster and more consistently for higher resolutions.

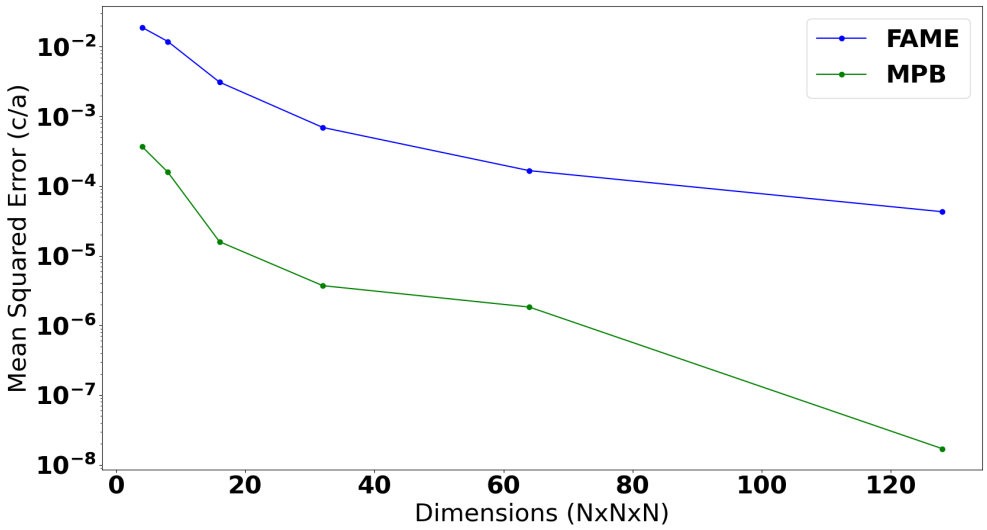

FIGURE 9: MSE of the eigenvalues for the sharp woodpile.

In Figure 9 we can see the error of the MPB runs decreases again at approximately the same rate but the FAME error is much higher than in Figure 8. Since FAME does seem to converge it is possible FAME is simply converging to a different band structure than MPB.

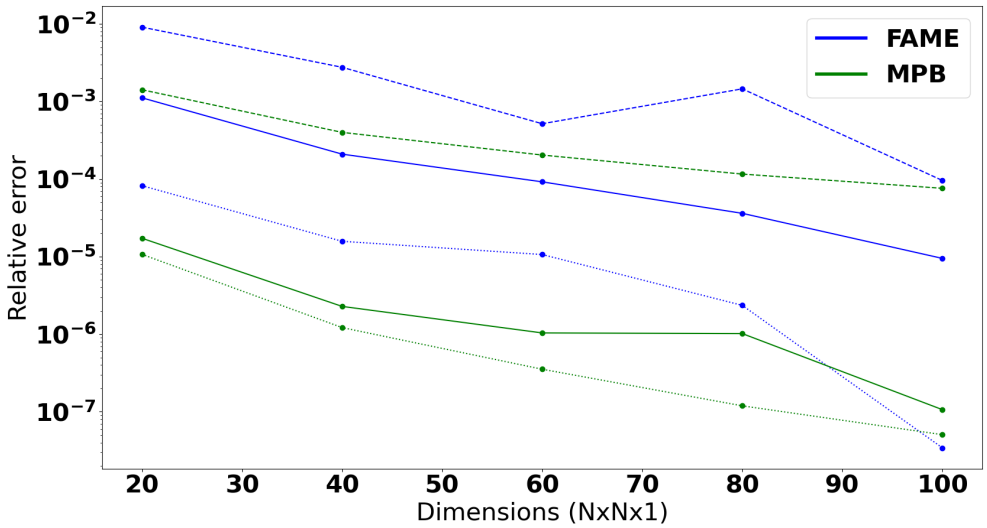

FIGURE 10: Relative error of the eigenvalues for the 2D crystal.

In Figure 10 we see that MPB does seem to have a downward trend, unlike in Figure 7. In fact, FAME seems to have a slightly higher relative error in general, much to the contrary of the results in Figure 7.

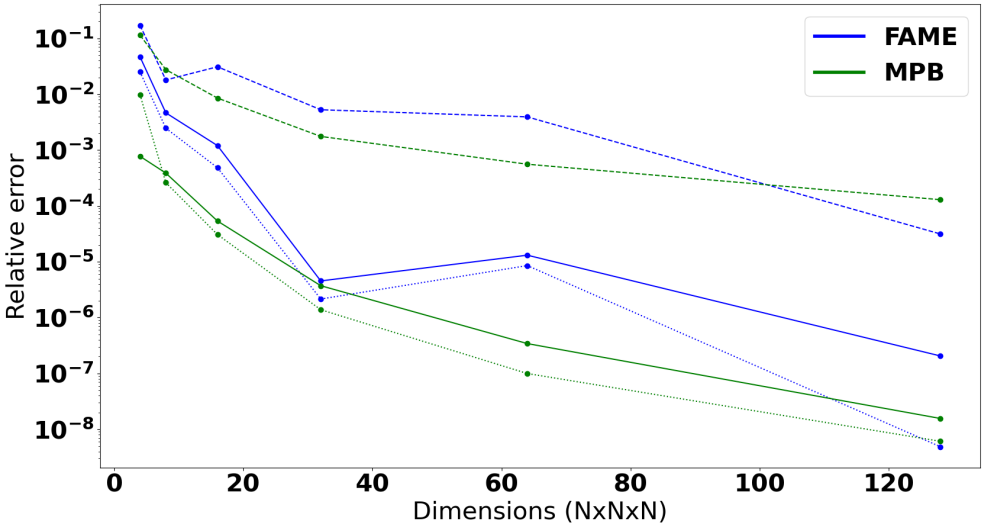

FIGURE 11: Relative error of the eigenvalues for the inverse woodpile.

In Figure 11 we see results very comparable to those of Figure 8. A general downward trend for both FAME and MPB and FAME consistently having slightly higher error which spikes at the $64 \times 64 \times 64$ resolution.

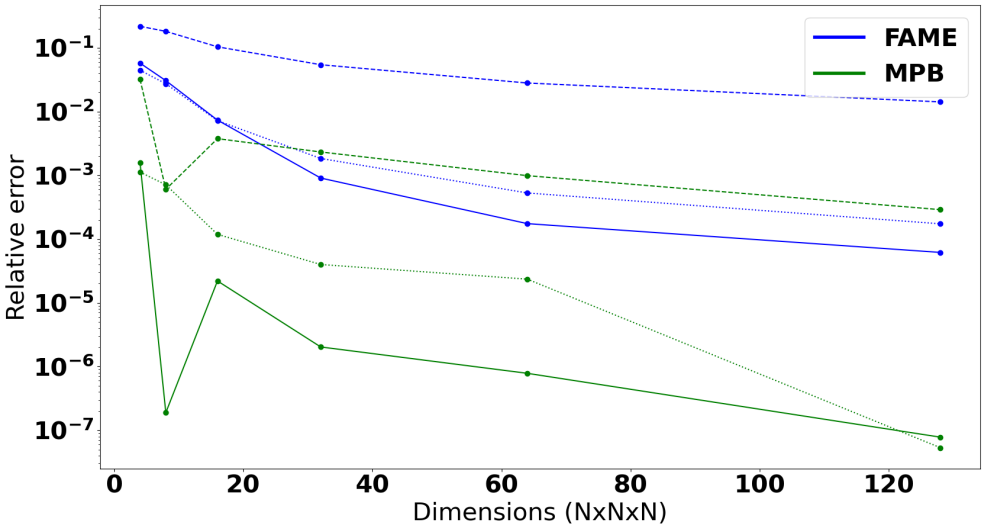

FIGURE 12: Relative error of the eigenvalues for the sharp woodpile.

In Figure 12 we see that much like in Figure 9 FAME consistently has a much higher error than MPB. This plot shows this difference is not because of one outlier but rather a trend among multiple of the frequency bands.

5 Discussion

There are several possible sources of error affecting that could affect the calculated data. As mentioned in Section 3, for the time measurements the outputs of the respective software packages were used to determine the required time, and MPB only outputs time in whole seconds. Thus, this lacking accuracy could be the cause for error in the average time per wave-vector. Similarly, the initialization time is only in whole seconds, which means the code could systematically be underestimating it, causing some of the constant difference. This could not cause the entire difference between FAME and MPB, however, since this would be responsible for at most $\frac{1}{|W|}$ seconds of error, while the constant difference between FAME and MPB in Figures 5 and 6 is clearly several seconds.

For the accuracy, firstly, the frequencies and electrical field produced by the high-resolution MPB simulation for reference values almost certainly vary from the exact value to a certain degree. As stated in Section 3, this error should be small compared to the measured error of the tested values, but it could misrepresent the error values. Secondly, since the reference values were generated using MPB there is a possibility that there is some systematic error introduced by MPB that does not depend on resolution, which artificially reduces the measured error in MPB's test values. Especially in the case of Figure 9 it is possible that the large error is caused by one of the packages improperly handling crystals with sharp edges, but even then it is unclear which of the packages is at fault. However, in this case it would be possible but unlikely that such an error would not depend on resolution, so since MPB attains such a low error, FAME is the more plausible candidate.

In order to implement the sharp woodpile crystal in FAME, some alterations to the source code were required, since by default only crystals that are unions of spheres and cylinders are supported. These alterations are presented in Appendix A. However, the fact that only spheres and cylinders are supported in the base code might be another reason for an optics researcher to prefer working with MPB over FAME. The FAME package has more such restrictions, such as the fact that although the FAME algorithm as described in [9] can calculate the frequency bands corresponding to any wave-vector, the FAME package can only calculate the band structure along the set of wave-vectors that make up the Brillouin path, and a configurable amount of wave-vectors in between. This is useful for plotting band structures, since this is exactly the set required to plot a band structure, but it is inconvenient when only a few of these wave-vectors are required, since calculations must be done for all of them. Furthermore, it means that the FAME package cannot calculate the frequency bands at a wave-vector not on the Brillouin path unless the user modifies the source code, even though the FAME algorithm is perfectly suitable for these wave-vectors. These issues do not reflect upon the FAME algorithm itself, but could be worth considering for researchers whose goals would require nontrivial modification of the source code for the FAME package, as in these cases MPB would likely be preferable.

6 Conclusion

From the results we can clearly see that FAME has a reduced computation time over MPB in most cases, but this effect wanes for higher resolutions. Furthermore, FAME generally produces a higher error than MPB, with only one exception. Especially given the magnitude of the additional error in the case of the sharp woodpile, when compared to the relatively small improvement in computing time, FAME is likely not a suitable replacement for MPB when working with high resolution simulations. However, due to the high relative performance improvement at low resolutions FAME could have valuable use in situations where many different simulations have to be run and the higher error from low resolutions is not an issue.

References

[1] Vivek Arjunan Vasantha et al. "Highly monodisperse zwitterion functionalized non-spherical polymer particles with tunable iridescence". In: *RSC Adv.* 9 (47 2019), pp. 27199–27207. DOI: 10.1039/C9RA05162G. URL: http://dx.doi.org/10.1039/C9RA05162G.

[2] Jong Bin Kim et al. "Designing Structural-Color Patterns Composed of Colloidal Arrays". In: *ACS Applied Materials & Interfaces* 11.16 (Apr. 2019), pp. 14485–14509. ISSN: 1944-8244. DOI: 10.1021/acsami.8b21276. URL: https://doi.org/10.1021/acsami.8b21276.

[3] Erik C. Nelson et al. "Epitaxial growth of three-dimensionally architectured optoelectronic devices". In: *Nature Materials* 10.9 (Sept. 2011), pp. 676–681. DOI: 10.1038/nmat3071.

[4] Steven G. Johnson and J. D. Joannopoulos. "Block-iterative frequency-domain methods for Maxwell's equations in a planewave basis". In: *Opt. Express* 8.3 (Jan. 2001), pp. 173–190. DOI: 10.1364/OE.8.000173. URL: http://www.osapublishing.org/oe/abstract.cfm?URI=oe-8-3-173.

[5] P Kano, D Barker, and M Brio. "Analysis of the analytic dispersion relation and density of states of a selected photonic crystal". In: *Journal of Physics D: Applied Physics* 41.18 (Aug. 2008), p. 185106. DOI: 10.1088/0022-3727/41/18/185106. URL: https://doi.org/10.1088/0022-3727/41/18/185106.

[6] Poonam Sharma and Preeta Sharan. "Design of Photonic Crystal-Based Biosensor for Detection of Glucose Concentration in Urine". In: *IEEE Sensors Journal* 15.2 (2015), pp. 1035–1042. DOI: 10.1109/JSEN.2014.2359799.

[7] E. N. Odarenko et al. "Analysis of Slow Wave Modes in Modified Photonic Crystal Waveguides Using the MPB Package". In: *2018 IEEE 17th International Conference on Mathematical Methods in Electromagnetic Theory (MMET)*. 2018, pp. 164–167. DOI: 10.1109/MMET.2018.8460468.

[8] S. Kuchinsky et al. "Coupling between photonic crystal waveguides". In: *IEEE Journal of Quantum Electronics* 38.10 (2002), pp. 1349–1352. DOI: 10.1109/JQE.2002.802954.

[9] Xing-long Lyu et al. “FAME: Fast Algorithms for Maxwell’s Equations for Three-Dimensional Photonic Crystals”. In: *ACM Trans. Math. Softw.* 47.3 (June 2021). ISSN: 0098-3500. DOI: 10.1145/3446329. URL: https://doi-org.ezproxy2.utwente.nl/10.1145/3446329.

[10] Kane Yee. “Numerical solution of initial boundary value problems involving maxwell’s equations in isotropic media”. In: *IEEE Transactions on Antennas and Propagation* 14.3 (1966), pp. 302–307. DOI: 10.1109/TAP.1966.1138693.

[11] Jon Häggblad and Olof Runborg. “Accuracy of staircase approximations in finite-difference methods for wave propagation”. In: *Numerische Mathematik* 128 (2014), pp. 741–771.

[12] Michael J. Mehl et al. “The AFLOW Library of Crystallographic Prototypes: Part 1”. In: *Computational Materials Science* 136 (2017), S1–S828. ISSN: 0927-0256. DOI: https://doi.org/10.1016/j.commatsci.2017.01.017. URL: https://www.sciencedirect.com/science/article/pii/S0927025617300241.

[13] Tsung-Ming Huang et al. *Solving Three Dimensional Maxwell Eigenvalue Problem with Fourteen Bravais Lattices*. 2018. arXiv: 1806.10782 [math.NA].

[14] Tsung-Ming Huang et al. “Eigendecomposition of the Discrete Double-Curl Operator with Application to Fast Eigensolver for Three-Dimensional Photonic Crystals”. In: *SIAM Journal on Matrix Analysis and Applications* 34 (Apr. 2013), pp. 369–391. DOI: 10.1137/120872486.

[15] L.A. Woldering et al. “The influence of fabrication deviations on the photonic band gap of three-dimensional inverse woodpile nanostructures”. Undefined. In: *Journal of Applied Physics* 105.093108 (2009), pp. 1–10. ISSN: 0021-8979. DOI: 10.1063/1.3103777.

[16] Xu Zheng et al. “Cavity Design in Woodpile Based 3D Photonic Crystals”. In: *Applied Sciences* 8 (July 2018), p. 1087. DOI: 10.3390/app8071087.

Appendix

A Code alteration for blocks in FAME

By default the FAME package does not support crystals that are not unions of spheres and cylinders. Although the algorithm outlined in [9] is completely compatible with other crystals, those who wish to simulate such crystals using FAME would have to alter the source code to add additional shapes. In order to add support for blocks so that the sharp woodpile can be simulated we have simply adjusted the normal test whether a point is contained in a cylinder to use the L_∞ norm ($\|v-w\| = \max\{|v_1-w_1|, |v_2-w_2|, |v_3-w_3|\}$) instead, essentially turning the cylinder into a block along the Cartesian axes. This new version of the code no longer supports ordinary cylinders, but the given function can easily be renamed and added alongside the old cylinder function. In this case the user would also have to add support for an additional type of shape in the internal data structures and in the parser for material data files.

```
int cylinder_handle(realCPU ptx, realCPU pty, realCPU ptz, realCPU* lattice_vec_a, MATERIAL Material, int kind)
{
    int i, sum = 0, flag = 0;
    int shiftx, shifty, shiftz;
    realCPU ctx, cty, ctz, cbx, cby, cbz;
    realCPU temp, distance;

    for(i = 0; i < kind; i++)
        sum += Material.cylinder_num[i];

    for(i = sum; i < sum + Material.cylinder_num[kind]; i++)
    {
        for(shiftx = -1; shiftx <= 1; shiftx++)
            for(shifty = -1; shifty <= 1; shifty++)
                for(shiftz = -1; shiftz <= 1; shiftz++)
                {
                    ctx = (Material.cylinder_top_centers[i * 3] + shiftx) * lattice_vec_a[0] + (Material.cylinder_top_centers[i
* 3 + 1] + shifty) * lattice_vec_a[3] + (Material.cylinder_top_centers[i * 3 + 2] + shiftz) * lattice_vec_a[6];
                    cty = (Material.cylinder_top_centers[i * 3] + shiftx) * lattice_vec_a[1] + (Material.cylinder_top_centers[i
* 3 + 1] + shifty) * lattice_vec_a[4] + (Material.cylinder_top_centers[i * 3 + 2] + shiftz) * lattice_vec_a[7];
                    ctz = (Material.cylinder_top_centers[i * 3] + shiftx) * lattice_vec_a[2] + (Material.cylinder_top_centers[i
* 3 + 1] + shifty) * lattice_vec_a[5] + (Material.cylinder_top_centers[i * 3 + 2] + shiftz) * lattice_vec_a[8];
                    cbx = (Material.cylinder_bot_centers[i * 3] + shiftx) * lattice_vec_a[0] + (Material.cylinder_bot_centers[i
* 3 + 1] + shifty) * lattice_vec_a[3] + (Material.cylinder_bot_centers[i * 3 + 2] + shiftz) * lattice_vec_a[6];
                    cby = (Material.cylinder_bot_centers[i * 3] + shiftx) * lattice_vec_a[1] + (Material.cylinder_bot_centers[i
* 3 + 1] + shifty) * lattice_vec_a[4] + (Material.cylinder_bot_centers[i * 3 + 2] + shiftz) * lattice_vec_a[7];
                    cbz = (Material.cylinder_bot_centers[i * 3] + shiftx) * lattice_vec_a[2] + (Material.cylinder_bot_centers[i
* 3 + 1] + shifty) * lattice_vec_a[5] + (Material.cylinder_bot_centers[i * 3 + 2] + shiftz) * lattice_vec_a[8];

                    temp = ((ptx - ctx) * (cbx - ctx) + (pty - cty) * (cby - cty) + (ptz - ctz) * (cbz - ctz)) / (pow(cbx - ctx,
 2) + pow(cby - cty, 2) + pow(cbz - ctz, 2));

                    if(0.0 <= temp && temp <= 1.0)
                    {
                        // changed portion:
                        distance = max(
                            max(abs(ctx + temp * (cbx - ctx) - ptx), abs(cty + temp * (cby - cty) - pty)),
                            abs(ctz + temp * (cbz - ctz) - ptz)
                        );
                        if(distance <= Material.cylinder_radius[i])
                        {
                            flag = 1;
                            break;
                        }
                    }
                }
    }
    return flag;
}
```