

# Accelerating Ultrasound Wave Propagation Simulations using Pruned FFT

1<sup>st</sup> Ondrej Olsak

Faculty of Information Technology  
Brno University of Technology  
Brno, Czech Republic  
Email: iolsak@fit.vut.cz

2<sup>nd</sup> Jiri Jaros

Faculty of Information Technology  
Brno University of Technology  
Brno, Czech Republic  
Email: jarosjir@fit.vut.cz

**Abstract**—The use of ultrasound in non-invasive medical procedures is a rapidly expanding area of medicine. The success of these treatments often depends on complex ultrasound simulations that require significant computing power, time, and associated calculation costs. To solve the differential equations associated with these simulations, the pseudo-spectral method with Fourier basis functions is employed. Thus, a significant part of the simulation time is spent computing the Fast Fourier Transform. This paper presents an approach that has the potential to reduce computation time and, consequently, the calculation costs of ultrasound wave propagation simulations used in the pre-planning phase of non-invasive treatments by involving the pruned Fast Fourier Transform algorithm (pruned FFT). The paper employs spectrum filtration using a binary map to emulate the behaviour of the pruned FFT. This allows for the evaluation of the impact of the pruned FFT on the number of computed elements in the spectral domain and the accuracy of the simulation. Results on real data have shown that it is possible to replace the Fast Fourier Transform (FFT) applied to acoustic pressure and velocity with the pruned version of the algorithm while obtaining results that are suitable for pre-planning purposes, thereby reducing computation time of the treatment planning. Involving the pruned FFT can also enable the execution of simulations in higher resolution domains with much faster execution times. In some cases, we were able to achieve around 90% accuracy on the single edge of the 2D domain.

**Index Terms**—Fourier transform, Simulation, Ultrasound, High performance computing

## I. INTRODUCTION

In medicine, ultrasound is used for diagnostic purposes, such as ultrasound imaging [1]. However, ultrasound can also be employed for non-invasive treatment purposes using focused high-intensity ultrasound. Focused high-intensity ultrasound can be used to produce temporary or permanent changes in brain tissue [2], making it suitable for non-invasive treatments. To achieve optimal transducer parameters and positioning, the k-Wave toolbox [3] utilises the pseudo-spectral method with Fourier basis functions [4]. Since the k-Wave toolbox employs Fourier basis functions, a significant portion of the simulation is consumed by computing the Fourier transform.

The most common algorithm to perform the Fourier transform is the Fast Fourier Transform (FFT) [5]. Some applications, however, may operate only over a subset of the spectral coefficients computed by the FFT algorithm. In such cases,

there may be a need to reduce computation time by computing only the required subset of the spectral coefficients through the use of a Sparse Fourier Transform (sparse FFT) [6] or Pruned Fourier Transform (pruned FFT) [7]. The difference between these two transforms lies in how they localise the desired coefficients in the spectrum. The sparse FFT suits signals with few non-zero coefficients at unknown spectral positions, typically using domain-specific knowledge [8]. Conversely, the pruned FFT algorithm is optimised for known zero patterns of the spectrum coefficients, bypassing unnecessary computations in the Fast Fourier Transform algorithm.

This paper aims to examine the impact of the pruned FFT when applied to acoustic pressure and velocity in two-dimensional ultrasound wave propagation simulations performed by the k-Wave toolbox [3]. Primary attention will be given to computational accuracy, the size of the computed area in the spectrum, and the position of the focus. The position of the ultrasound focus is crucial since a specific position in the tissue needs to be targeted. Whilst the Sparse FFT was also considered, earlier experiments have shown that this algorithm is not suitable for this kind of computation [9].

In the following sections, the mathematical definition of the ultrasound wave propagation simulation implemented in k-Wave will be described, followed by a section detailing the experimental pipeline that allows the area in the spectrum of acoustic pressure and velocity for the pruned FFT to be estimated using the Acoustic Field Propagator (AFP) [10]. Finally, the impact of applying the pruned FFT on simulation accuracy will be evaluated. The results from this paper should serve as a theoretical basis for the future deeper evaluation of the effect of the pruned FFT on ultrasound wave propagation simulation.

## II. MATHEMATICAL BACKGROUND

To compute ultrasound wave propagation, the *k-Wave* toolbox employs the pseudo-spectral method using Fourier basis functions. This technique involves representing the solution of the differential equation as a sum of specific basis functions. Unlike finite-difference time domain methods, which rely on local computations at neighbouring points, spectral methods utilise information from the entire domain, leading to higher computational accuracy [4].

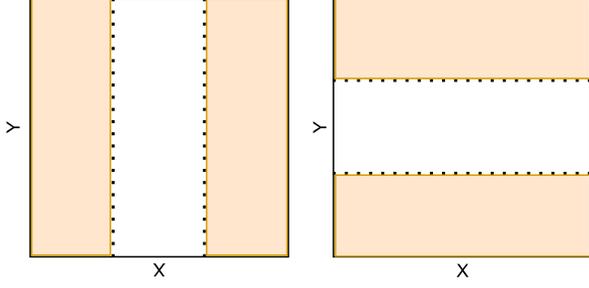


Fig. 1: Illustration of the filter orientation.

The k-Wave toolbox runs simulations based on the following governing equations [3]:

$$\begin{aligned}
 \frac{\partial u}{\partial t} &= -\frac{1}{\rho_0} \nabla p \\
 \frac{\partial \rho}{\partial t} &= -\rho_0 \nabla \cdot u - u \cdot \nabla \rho_0 \\
 p &= c_0^2 \left( \rho + d \cdot \nabla \rho_0 + \frac{B}{2A} \frac{\rho^2}{\rho_0} - L_\rho \right)
 \end{aligned} \quad (1)$$

Equation (1) can be written in a discrete form using the  $k$ -space pseudo-spectral method [11]. This equation is part of the spatial gradient calculations based on the Fourier collocation spectral method.

$$\frac{\partial}{\partial \xi} p^n = \mathcal{F}^{-1} \{ i k_\xi \kappa e^{i k_\xi \xi} \mathcal{F} \{ p^n \} \} \quad (2)$$

In Equation (2), for the Cartesian direction  $\xi = x$  in  $R^1$ ,  $\xi = x, y$  in  $R^2$ ,  $\mathcal{F}$  and  $\mathcal{F}^{-1}$  denote the forward and inverse spatial Fourier transform,  $i$  is the imaginary unit,  $k_\xi$  represents the wave numbers in the  $\xi$  direction, and  $\kappa$  is the  $k$ -space operator defined as  $\kappa = \text{sinc}(c_{ref} k \Delta t / 2)$ , where  $c_{ref}$  is a scalar reference sound speed.

The Fast Fourier Transform (FFT) algorithm is used to convert signals from the spatial domain to the spectral domain. Each simulation step of the ultrasound propagation simulation involves 14 FFTs in the 3D domain and 11 FFTs in the 2D domain. This computation consumes approximately 60% of the total simulation time, making it a significant part of the overall simulation [12].

### III. EMULATING PRUNED FFT WITH FILTRATION

To emulate the pruned FFT in the ultrasound wave propagation performed by k-Wave, filtration using a 2D binary map was employed. In each simulation step, the full FFT over acoustic pressure and velocity is computed. The spectrum is multiplied by the binary map representing the coefficients of the pruned FFT. The result of this multiplication is analogous to the spectrum computed by the pruned FFT, as coefficients outside the filter become zero through multiplication.

The changes to the original simulation source code lie in the modified HDF5 input file, which holds all datasets and attributes. The file was extended with a so-called 'filtration

kernel', which is a 2D dataset holding the filtration binary map that is used to multiply the resulting FFT spectrum. The filter always has one edge equal to the size of one dimension's edge based on its orientation. Figure 1 illustrates both possible filter orientations. Firstly, the reduced dimension is X (columns), and secondly, the reduced dimension is Y (rows). The orange area represents ones, and the white represents zeros. We primarily target low frequencies; therefore, the filter is placed on the sides of the domain. By computing the full row/column of the coefficients, better accuracy is achieved. Moreover, this approach may lead to a simpler implementation of the pruned FFT in the future. Although it is possible to use the symmetry of the complex domain where only half of the coefficients are computed, for better understanding, all row and column sizes in the experiments will be given in full domain sizes. The percentage of computed or skipped rows/columns remains the same for both reduced and full dimension sizes.

### IV. EXPERIMENTAL PIPELINE

To automate the process of simulation input file creation, spectrum area estimation, simulation, and result evaluation, the *experimental pipeline* was created. The scheme of the pipeline with the flow of all HDF5 files can be seen in Figure 2. The pipeline consists of the following five steps:

**Input files creation** - Creation of the k-Wave input file includes specifying all simulation properties such as media sound speed, density, transducer shape and frequency, grid spacing, and simulation time. These simulation properties are also used to create an input file for the Acoustic Field Propagator (AFP) [10]. The AFP input file contains transducer properties, domain properties, and the time at which the pressure field will be computed. All properties are inherited from the k-Wave simulation, except the time. Since the AFP only works with homogeneous media, the lowest sound speed value from the original simulation is used, as this results in higher frequencies in the spectrum. The simulation time is chosen to capture the propagated wave before it reaches the edge of the domain in at least one direction, ensuring the transition between the wave and the steady part of the domain is obtained.

**Acoustic field propagation** - In this step, the calculation of the acoustic field from single-element and phased array transducers in homogeneous media is performed using the AFP. The AFP enables the calculation of the acoustic pressure field at all spatial positions at a given time in a single step. The computation time is significantly lower compared to the full ultrasound propagation simulation. However, this method cannot compute propagation in heterogeneous media or the reflection of the propagated wave, thus it cannot replace a full ultrasound wave propagation simulation. Despite its limitations, it provides an acceptable estimation of the spectral coefficient positions.

**Spectrum area estimation** - To estimate the area for the filtration, the acoustic pressure from the previous step is used. The bisection approach is employed, where the condition is in the form of the error in the space domain. This provides

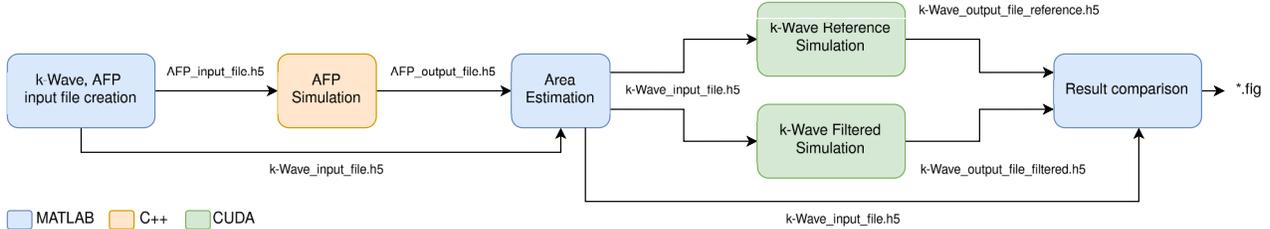


Fig. 2: The evaluation pipeline to estimate area in the spectral domain and compare the result of the reference and filtered version of the simulation.

an approximate position of the most significant coefficients in the spectrum. Since most of the significant coefficients are low frequencies [9], the spectrum is shifted for the purpose of the bisection, placing all low frequencies around the centre of the spectral domain. In each iteration, the coefficients outside the borders are temporarily zeroed, and the inverse FFT is computed. The resulting space domain is compared to the original one, and based on the error, the borders in the spectral domain are shifted towards (if the error is below the threshold) or away from (if the error is above the threshold) the centre of the domain. This operation is symmetrical about the centre of the domain. The final filter area is added to the k-Wave simulation file in the form of the binary filter.

**Ultrasound wave propagation simulation & Evaluation -** In this step, two simulations are executed. Firstly, the original ultrasound wave propagation and secondly, its modified version with acoustic pressure and velocity filtration. To compute both simulations, the CUDA implementation of the simulation is used. Finally, both the results of the original and filtered simulations are compared. The figures with the reference and filtered acoustic pressure, their spectrum in the last simulation step, and the normalised  $L_\infty$  error in the space and spectrum domains are created.

#### A. Estimation of the filtration direction

The direction in which the wave propagates together with transducer properties affect the position of the coefficients in the spectral domain [13]. The wave propagation direction is primarily given (among other factors) by the transducer position. In Figure 3, two positions of the piston transducer were used in ultrasound wave propagation in water. It can be seen that the coefficients in their spectra are in different positions. To capture the significant coefficients while reducing the number of computed coefficients, bisection is performed over both directions—rows and columns. Since the error threshold is the same in both cases, the direction of the filter is chosen based on the smaller number of rows/columns that need to be computed.

Table I shows the impact of the filtration direction on the size of the filter area in the simulation shown in Figure 3. It can be observed that the direction of the filter impacts the accuracy of the computation, though not as significantly

TABLE I: Effect of the filtration direction on the number of computed rows/columns and error.

Transducer alignment	Edge size	Reduced dimension	Row/Column Skip cnt.	Skip [%]	$L_{inf}$ [%]
X	1024	X	636	62.11	1.71
Y	1024	X	42	4.10	0.18
X	1024	Y	42	4.10	0.20
Y	1024	Y	636	62.11	1.73

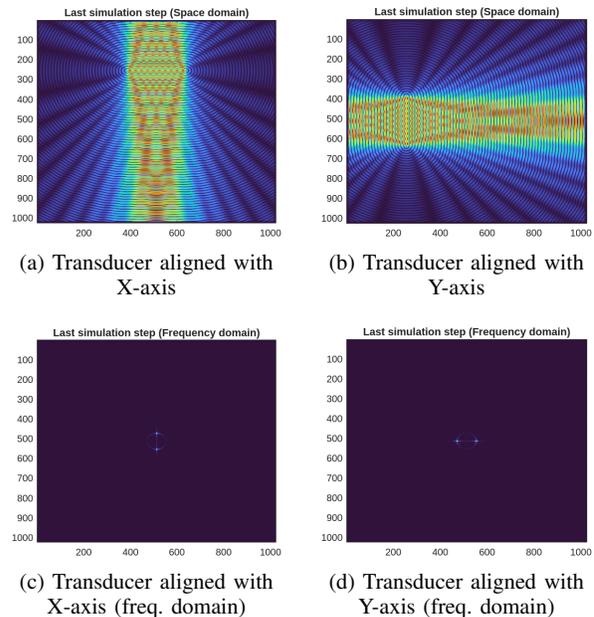


Fig. 3: Effect of the transducer position on the frequency domain

as the number of computed rows/columns. In all subsequent experiments, the filter direction will be chosen based on the number of rows/columns that need to be computed.

#### B. Estimation of the coefficient area

The bisection algorithm is driven by the error in the space domain computed after the spectral domain coefficient filtration over the result of the AFP. Three possible errors were considered: Mean Absolute Percentage Error (MAPE), Root

TABLE II: Comparison of error and skipped rows/columns under different levels of Mean Absolute Percentage Error (MAPE) in domain with edge size of 1024.

MAPE	10%	20%	30%	40%	50%
<b>Rows skip</b>	1	502	552	712	760
<b>Rows skip [%]</b>	0.10	49.02	53.91	69.53	74.22
$L_\infty$ [%]	0	0.51	0.59	1	1.24

TABLE III: Comparison of error and skipped rows/columns under different levels of Normalised Percentage  $L_\infty$  error in domain with edge size of 1024.

Norm. $L_\infty$	1%	2%	3%	4%	5%
<b>Rows skip</b>	670	810	856	886	906
<b>Rows skip [%]</b>	65.43	79.10	83.59	86.52	88.48
$L_\infty$ [%]	0.85	1.76	2.45	3.44	4.73

Mean Squared Percentage Error (RMSPE), and Normalised Percentage  $L_\infty$  Error. However, during the experiments, the RMSPE proved to be inappropriate for the purpose of the bisection. Tables II and III show the results of the coefficient filtration with bisection using MAPE and Normalised Percentage  $L_\infty$  Error in a simulation with an arc transducer in water aligned with the X axis (filtered dimension is X).

The results show that Normalised Percentage  $L_\infty$  error is much more predictable in the case of the final simulation error, where the selected threshold is, in the simplest simulation, always higher than the final error. Subsequent experiments will show that this may not always be true. Another advantage is its ability to include significant coefficients in the filtration area. If there is a coefficient in the spectrum that would significantly affect the result in the space domain, we are able to include it in the computed area much more precisely than by using MAPE, as the error represents the point with the highest difference from the reference simulation. In contrast, MAPE represents the error across all points in the domain. The disadvantage of the MAPE error is also its lack of intuitiveness for possible bisection threshold adjustment. The Normalised Percentage  $L_\infty$  Error is much more suitable for the area estimation, and thus we will use it in all subsequent experiments.

## V. EVALUATION OF THE METHOD OVER REAL DATA

In this section, the impact of filtration on the spectrum will be evaluated with focus on spectrum coefficient reduction, position of the focus and the overall simulation error. The error in the domain will be expressed in the form of the Normalised Percentage  $L_\infty$  error [14]. To evaluate the impact of the pruned FFT over real data, all experiments will be measured on ultrasound wave propagation in the human head. To show the effect of transducer position on the simulation result, the following experiments will contain two positions of piston and arc transducer: one aligned with the X axis (Figure 4a) and the second unaligned with either axis (Figure 4b).

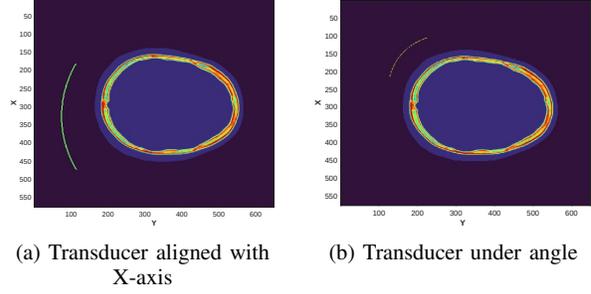


Fig. 4: Position of the transducer to the head used in the experiments.

The piston transducer will be placed in the same position as the arc transducer in Figure 4. The dot transducer will be measured only once, as it cannot be "aligned" with the domain axis. The frequency of the transducer was set to 600kHz with amplitude equal to 10Pa in all experiments. The original domain size was  $576 \times 648$  grid points, with uniform grid spacing of  $4.6875 \times 10^{-4}$ , time step  $dt = 53.2181ns$ , and  $CLF = 0.3$ . This leads to 5.3 points per wavelength (in water). The simulation ran for 5061 steps, based on wave travel time across the diagonal of the grid. The first set of experiments focuses on increased resolution of the simulation. To correctly represent the geometry of the tissue, especially bones and the skull, a higher resolution of the simulation may be required [15]. The resolution of the tissue geometry impacts the error in the phase of the wave. Therefore, the higher the simulation resolution, the more precise the simulation will be. Since only the grid spacing and the number of grid points will change, the spectral domain of such a simulation will contain more zero or negligible coefficients. This should lead to an increase in the number of filtered coefficients in the spectrum.

In the first set of experiments, the domain resolution was progressively increased by a factor of two, starting from the original size up to 8 times the original domain size, where the grid spacing reduces by the same factor to keep the physical size of the domain constant. Table IV shows the results of the simulations over various resolutions with 2% and 3% bisection error thresholds on different transducer types and positions (X in the name of the experiment represents transducer alignment). The chosen filter dimension in all cases was X-axis, so it is omitted in the table. It can be observed that the position of the transducer has a significant impact on the number of computed rows in both arc and piston transducers. The bisection threshold affects not only the number of filtered rows but also the calculation error, as it allows the bisection algorithm to further reduce the calculated area. Furthermore, the final error in some cases does not match the given threshold error, as seen in the simulation in water. This is caused by the heterogeneity of the media.

Figure 5 shows the Normalised  $L_\infty$  error for the ArcX experiment with a 3% bisection error. As we can see, the error mostly occurs at the boundary between two media with a large difference in sound speed properties (e.g., skin →

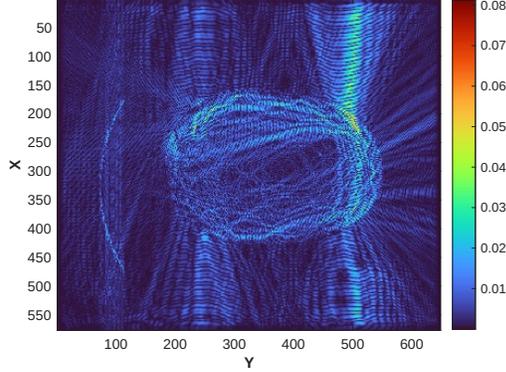


Fig. 5: The Normalised  $L_\infty$  error for ArcX experiment.

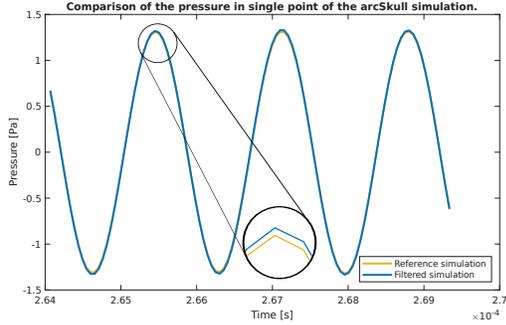


Fig. 6: Focal point pressure variation in final 100 steps of ArcX simulation.

skull). However, there is no shift in the focal point, and its error is around 3%. Figure 6 shows the value of the pressure at the focal point during the last 100 simulation steps. In the case of the Piston and PistonX experiments, the results are close to the Arc and ArcX experiments, where the position of the transducer affects the number of filtered coefficients. In the case of the Piston, the overall computation error is significantly better due to the geometry of the piston transducer. The Dot experiments showed that this approach is not suitable for this type of transducer. The coefficient reduction is quite low and the simulation error is relatively high compared to the experiments with other transducer types. Increasing the resolution improves the number of computed rows and reduces simulation error. When the domain size of the simulation is increased while maintaining the same grid spacing, the physical size of the simulation expands. The next set of experiments was performed on such a domain. The same sizes as in the increased resolution were used, but the grid spacing remained unchanged. Table V shows that increasing the physical size of the domain leads to an increase in error, especially in the ArcX experiment. In the Dot, PistonX, and Piston experiments, the values remain relatively consistent across each domain size.

Our results show that coefficient reduction depends on transducer positioning. While the dot transducer cannot be aligned with domain edges, both piston and arc transducers can

be aligned through domain rotation if necessary. After rotation, the rectangular shape of the domain can be restored by filling the domain with the surrounding media. Since the domain is surrounded by a Perfect Match Layer [3], no reflection will affect the simulation results.

## VI. DISCUSSION

When we compare other tools for solving acoustic pressure wave equations [14], we can see that for some benchmarks, the median values for the cross-comparison range between 10% and 100% for the  $L_\infty$  error. It is difficult to compare our results directly with those benchmarks, as our simulations differ from those presented in [14]. However, computation errors achieved in our set of experiments are promising, especially for high-resolution simulations.

The method is limited in maintaining high frequencies that represent edges in spatial domain. Thus, in cases of two media with significantly different sound speeds, this approach may not give correct results.

Regarding acceleration, approximately 60% of the simulation computation is dedicated to calculating the Fourier transform. If we compute roughly 60% of the rows/columns, we could achieve an acceleration of around 30%. Since multiplication of the zero coefficients equals zero, there will be additional reduction in other mathematical operations, especially in the spectral domain. Due to the lower number of coefficients in the spectral domain, there will also be a reduction in memory accesses. Depending on the future implementation, the sizes of the matrices holding the results of the pruned FFT could potentially be reduced to match the size of the nonzero elements, leading to a memory reduction. In 3D simulation the reduction will be even more significant since another dimension will be reduced.

## VII. CONCLUSION

The integration of the pruned FFT algorithm into ultrasound wave propagation simulations shows potential for improving computational efficiency, particularly in the pre-planning phase of a medical treatments. Our experiments demonstrate that it is possible to apply spectrum filtration using a binary map to emulate the pruned FFT, thus reducing the number of computed elements in the spectral domain and lowering computation time with acceptable accuracy of the simulations. This approach is particularly beneficial when working with high-resolution domains, where the computational time may be significantly greater.

The experiments have shown that the position of the transducer has a significant impact on computation accuracy and the number of computed coefficients. By optimising the transducer's position and the error thresholds in the bisection algorithm, we were able to achieve substantial reductions in the number of computed rows while maintaining acceptable levels of simulation error. The results indicate that the pruned FFT can be a powerful tool for reducing the computational load of ultrasound simulations. The best results were achieved in high-resolution simulations, where the number of reduced

TABLE IV: The measurement for the various resolution of the original domain for 2% and 3% threshold error.

	2%								3%							
	1x		2x		4x		8x		1x		2x		4x		8x	
	Skip [%]	$L_\infty$ [%]														
Arc	19.10	1.68	22.22	0.33	22.66	0.16	22.79	0.11	28.12	2.64	33.16	0.61	33.94	0.31	34.16	0.12
ArcX	46.18	2.26	72.22	1.47	82.81	1.56	90.10	3.02	57.29	8.12	76.39	3.10	87.07	3.53	92.97	4.84
Piston	3.82	0.73	4.17	0.24	4.17	0.14	4.17	0.15	5.90	0.87	6.08	0.22	6.25	0.13	6.25	0.15
PistonX	19.44	1.14	35.59	0.49	56.25	0.41	75.87	0.48	26.39	1.71	46.35	0.61	68.40	0.64	80.86	1.16
Dot	6.25	3.74	8.51	1.02	10.59	1.44	12.67	3.27	9.03	4.70	12.33	1.24	15.28	1.64	18.06	3.37

TABLE V: The measurement for the various domain sizes of the original domain for 2% and 3% threshold error.

	2%								3%							
	1x		2x		4x		8x		1x		2x		4x		8x	
	Skip [%]	$L_\infty$ [%]														
Arc	19.10	1.68	27.60	1.24	35.33	1.67	44.31	3.19	28.12	2.64	38.37	2.03	45.49	2.65	52.34	4.74
ArcX	46.18	2.26	58.68	5.20	62.50	17.44	64.89	79.78	57.29	8.12	62.67	28.99	67.01	53.21	72.92	86.07
Piston	3.82	0.73	3.82	0.39	3.91	0.23	3.95	0.23	5.90	0.87	5.73	0.39	5.82	0.29	5.90	0.32
PistonX	19.44	1.14	19.62	0.44	19.27	0.34	19.49	0.40	26.39	1.71	26.91	0.64	26.48	0.54	26.74	0.58
Dot	6.25	3.74	6.25	3.27	6.25	1.62	6.25	3.38	9.03	4.70	9.20	4.33	9.11	2.26	9.11	4.92

coefficients was up to 90%, with approximately 5% error. Future research will focus on the implementation of the pruned FFT algorithm and its integration into k-Wave ultrasound wave propagation simulation.

#### ACKNOWLEDGMENT

This work was supported by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90254). This project has received funding from the European Unions Horizon Europe research and innovation programme under grant agreement No 101071008. This work was supported by Brno University of Technology under project number FIT-S-23-8141.

#### REFERENCES

- [1] C. Shou, X. Chen, H.-L. Liu, and P.-H. Tsui, "Using short-time fourier transform to ultrasound signals for fatty liver detection," *International Journal of Signal Processing Systems*, pp. 300–303, 08 2016.
- [2] W. J. Fry and F. J. Fry, "Fundamental neurological research and human neurosurgery using intense ultrasound," *IRE Transactions on Medical Electronics*, vol. ME-7, no. 3, pp. 166–181, 1960.
- [3] E. B. Treeby, J. Jaros, P. A. Rendell, and T. B. Cox, "Modeling nonlinear ultrasound propagation in heterogeneous media with power law absorption using a k-space pseudospectral method," *Journal of the Acoustical Society of America*, vol. 131, no. 6, pp. 4324–4336, 2012.
- [4] S. Gottlieb and D. Gottlieb, "Spectral methods," *Scholarpedia*, vol. 4, no. 9, p. 7504, 2009, revision #91796.
- [5] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex fourier series," *Mathematics of Computation*, vol. 19, no. 90, pp. 297–301, 1965. [Online]. Available: <http://www.jstor.org/stable/2003354>
- [6] J. Schumacher and M. Püschel, "High-performance sparse fast fourier transforms," in *2014 IEEE Workshop on Signal Processing Systems (SiPS)*, 2014, pp. 1–6.
- [7] J. Markel, "Fft pruning," *IEEE Transactions on Audio and Electroacoustics*, vol. 19, no. 4, pp. 305–311, 1971.
- [8] H. Hassanieh, F. Adib, D. Katabi, and P. Indyk, "Faster gps via the sparse fourier transform," in *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, ser. Mobicom '12. New York, NY, USA: Association for Computing Machinery, 2012, p. 353–364. [Online]. Available: <https://doi.org/10.1145/2348543.2348587>
- [9] O. Olsak and J. Jaros, "Techniques for efficient fourier transform computation in ultrasound simulations," in *Proceedings of the 33rd International Symposium on High-Performance Parallel and Distributed Computing*, ser. HPDC '24. New York, NY, USA: Association for Computing Machinery, 2024. [Online]. Available: <https://doi.org/10.1145/3625549.3658825>
- [10] B. E. Treeby, J. Budisky, E. S. Wise, J. Jaros, and B. T. Cox, "Rapid calculation of acoustic fields from arbitrary continuous-wave sources," *The Journal of the Acoustical Society of America*, vol. 143, no. 1, pp. 529–537, 01 2018. [Online]. Available: <https://doi.org/10.1121/1.5021245>
- [11] B. Treeby, B. Cox, and J. Jaros, "k-wave a matlab toolbox for the time domain simulation of acoustic wave fields user manual," 2016. [Online]. Available: [http://www.k-wave.org/manual/k-wave\\_user\\_manual\\_1.1.pdf](http://www.k-wave.org/manual/k-wave_user_manual_1.1.pdf)
- [12] J. Jaros, B. Treeby, and A. Rendell, "Use of multiple gpus on shared memory multiprocessors for ultrasound propagation simulations," vol. 127, 01 2012, pp. 43–52.
- [13] M. Gircys and B. J. Ross, "Image evolution using 2d power spectra," *Complexity*, vol. 2019, no. 1, p. 7293193, 2019. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1155/2019/7293193>
- [14] J.-F. Aubry, O. Bates, C. Boehm, K. Butts Pauly, D. Christensen, C. Cueto, P. G lat, L. Guasch, J. Jaros, Y. Jing, R. Jones, N. Li, P. Marty, H. Montanaro, E. Neufeld, S. Pichardo, G. Pinton, A. Pulkkinen, A. Stanzola, A. Thielscher, B. Treeby, and E. van 't Wout, "Benchmark problems for transcranial ultrasound simulation: Intercomparison of compressional wave models," *The Journal of the Acoustical Society of America*, vol. 152, no. 2, p. 1003–1019, Aug. 2022. [Online]. Available: <http://dx.doi.org/10.1121/10.0013426>
- [15] J. L. B. Robertson, B. T. Cox, J. Jaros, and B. E. Treeby, "Accurate simulation of transcranial ultrasound propagation for ultrasonic neuromodulation and stimulation," *The Journal of the Acoustical Society of America*, vol. 141, no. 3, pp. 1726–1738, 03 2017. [Online]. Available: <https://doi.org/10.1121/1.4976339>