GPU Accelerated Acoustic Field Determination for a Continuously Excited Circular Ultrasonic Transducer

Alberto Lemos Duran [*,1] André K. Sato [**,2]
Agesinaldo M. Silva Jr [*,3] Ediguer E. Franco [***,4]
Flávio Buiochi [*,5] Thiago C. Martins [**,6]
Júlio C. Adamowski [*,7] Marcos S. G. Tsuzuki [**,8]

[*] *Laboratory of Sensors and Actuators*
[**] *Laboratory of Computational Geometry*
Mechatronics and Mechanical Systems Engineering Department,
Escola Politécnica da Universidade de São Paulo, São Paulo, Brazil

[***] *Engineering Faculty, Universidad Autónoma de Occidente*
Cali, Colombia

Abstract:
This work presents a GPU algorithm to calculate the acoustic field generated by a circular ultrasonic transducer radiating in water a continuous wave. The acoustic pressure in a space point in front of the transducer is calculated by Rayleigh integral, which uses the Huygens principle to compose the field as the sum of contributions from an infinite number of point sources. Because the pressure at each spatial point can be calculated independently, the solution algorithm can run in parallel, taking advantage of the GPU cores. Some experiments were performed in a frequency range from 0.25 to 5.0 MHz. The radiating surface was discretized in order to have a fixed number of elemental areas per wavelength. Results showed the validity of the acoustic fields simulated. In addition, a performance analysis showed that the GPU was 50 times faster than CPU for the most demanding problems. *Copyright ©2020 IFAC.*

Keywords: Ultrasound, Acoustic Field, GPU.

1. INTRODUCTION

The application of ultrasound in the industry has been increasing fast (Adamowski et al., 2013). It is related to rapid advances in transducers, data acquisition, signal processing and others. The acoustic field determination is an important feature when developing applications for ultrasound. It provides information on how sound waves are propagated and how energy is distributed in the medium (Treeby and Cox, 2010).

The acoustic field can be determined by solving the Rayleigh integral, which considers the interaction between the planar generating source and the propagation field generated by a baffled piston. This problem can consider a harmonically excited transducer or a different form of excitation (transient analysis).

Although most of studies consider a uniform emitting surface displacement, this does not occur in real situations and functions for displacement can be introduced (He and Hay, 1993). There is analytic solution for the axial pressure field produced by harmonically excited circular transducers (Kinsler et al., 1999), the King integral solution for the pressure field produced by harmonically excited circular transducers (Greenspan, 1979), numerical solutions for the pressure field produced by circular transducers excited by generic functions (pulse response) (Weight, 1984; Franco et al., 2011; San et al., 2006) and a method to approach the spatial pulse response produced by an arbitrary planar source (Piwakowski and Delannoy, 1989). As most transducers are circular, most simulations use simple geometries; however there are specific applications in which it is necessary to use more complex geometries (e.g. in acoustic vortexes) and the Rayleigh integral method allows it.

Acoustic field simulation is a very demanding task. Depending on the size of the analysis domain, the operation frequency of the transducer and the physical assumptions involved, the computational load and problem complexity increase such that acoustic field simulations take longer.

The computing power of GPUs has increased dramatically in the last decade due to high processing parallelism and high memory band width developed to meet demand for realtime and high definition 3D graphics (Nvidia, 2018). This computing power has been used for general-purpose

[1] e-mail: duran@usp.br.
[2] e-mail: andre.kubagawa@gmail.com.
[3] e-mail: agesinaldo.jr@gmail.com.
[4] e-mail: eefranco@uao.edu.co.
[5] e-mail: fbuiochi@usp.br.
[6] e-mail: thiago@usp.br.
[7] e-mail: jcadamow@usp.br.
[8] e-mail: mtsuzuki@usp.br.

programming in many areas of engineering and science. Literature report recent works related to the use of GPUs in astronomy (Wang et al., 2015), fluid dynamics (Franco et al., 2015; Tsiakas et al., 2019), big data (Johnson et al., 2019), precision agriculture (Patrício and Rieder, 2018), medical imaging (Lebedev et al., 2018), among many others. In the specific case of simulations in acoustics, several authors used GPU to accelerate the solution of the ultrasound wave equation (Mehra et al., 2012; Diaz et al., 2018; Wang et al., 2018).

In this work, a GPU-based solution is proposed herein to accelerate the acoustic field simulations and improve the implementation performance. The approach proposed here is different, as it is considered a harmonically excited circular transducer.

The text is structured as follows: Section 2 describes the problem, and Section 3 contains the proposed improved parallel solution. In Section 4, the experimental results are reported and discussed. Finally, conclusions are drawn in Section 5.

2. PROBLEM DESCRIPTION

In this work, the acoustic field is determined by the Rayleigh integral method (Kirkup, 1994). It uses the Huygens principle to compose the field generated by a planar source by adding contributions from an infinite number of point sources. Then, for a point Q in the propagation field, the pressure is given by

$$P_Q(r_Q, t) = \frac{j\omega\rho}{2\pi}\int_S \frac{e^{-jkr}}{r}\cdot \nu_n(x_s, y_s, t)dS \qquad (1)$$

where:

- t: time instant;
- ω: transducer angular frequency;
- ρ: propagation medium density;
- r: distance between Q and the differential area (dS);
- r_Q: position vector of point Q. The position is defined as a vector starting at the center of the circle;
- $\nu_n(x_s, y_s, t)$: normal component of the transducer velocity for point (x_s, y_s) at instant t;
- k: wavenumber.

For an harmonic excitation, the normal component of the transducer velocity is harmonic of the form $\nu_n(x_s, y_s, t) = \nu_n(x_s, y_s)e^{j\omega t}$. The response for a uniformly distributed vibration amplitude is

$$P_Q(r_Q) = \frac{j\omega\rho U_0}{2\pi}\int_S \frac{e^{-jkr}}{r}dS, \qquad (2)$$

where U_0 is the uniform normal component of the transducer velocity.

By dividing the transducer surface domain into small squares, which are approximated as point sources, as shown in Fig 1, the pressure is given by

$$\tilde{P}_Q(r_Q) = \frac{j\omega\rho U_0 dA}{2\pi} \left\{\sum_{n=1}^{N}\left[\frac{\cos(kr_n)}{r_n}\right] - j\sum_{n=1}^{N}\left[\frac{\sin(kr_n)}{r_n}\right]\right\} \qquad (3)$$

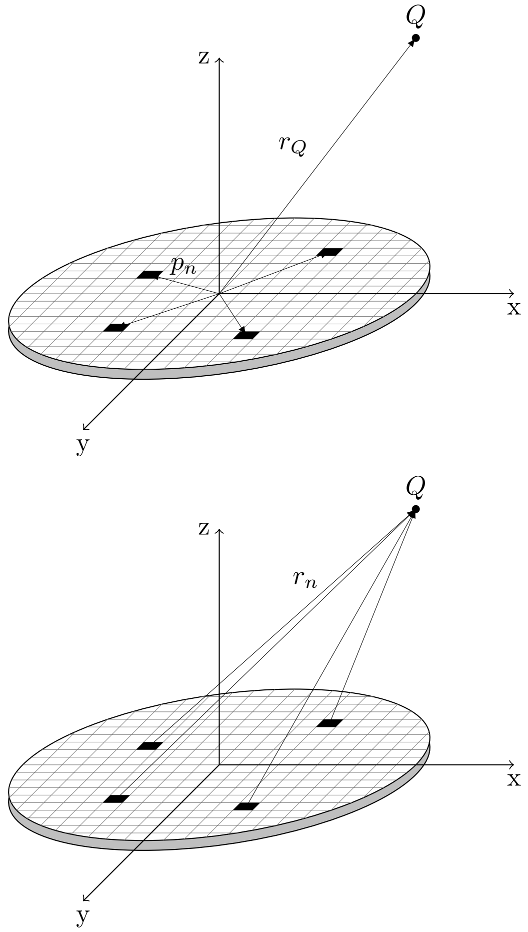


Fig. 1. Transducer surface discretization in N elements, for the Rayleigh integral method. Vector r_Q determines the position of point Q and vector p_n represents the position of the differential square area, both consider the center of the circle as reference point. Vector r_n represents the path for the acoustic wave starting at the differential square area dA and ending at point Q.

where r_n is the distance between the differential area and point Q. Finally, by adopting the parameter pps, which dictates the number of divisions per wavelength λ, to control the discretization as $dx = dy = \frac{\lambda}{pps}$, the absolute pressure of point Q is

$$\left\|\tilde{P}_Q(r_Q)\right\| = \frac{\rho U_0 \lambda^2 f}{pps^2} \left\|\sum_{n=1}^{N}\left[\frac{\cos(kr_n)}{r_n}\right] - j\sum_{n=1}^{N}\left[\frac{\sin(kr_n)}{r_n}\right]\right\|. \qquad (4)$$

3. PROPOSED GPU ACCELERATED SOLUTION

The numerical integration technique discretizes the domain, which includes the transducer and the volume of interest in its vicinity. Using the Huygens principal, the absolute pressure for each point is given by eq. (4). To allow more paralellization, the pressure determination can be divided into two steps. The first consists of determining the two sums, which derives from the real and imaginary parts of the pressure. Then, the absolute pressure is obtained by computing the norm of the resulting complex number and multiplying it by a scalar.

Algorithm 1 Absolute pressure determination kernel

```
1: $r_x \leftarrow r_{Q,x}[tid/2]$; $r_y \leftarrow r_{Q,y}[tid/2]$; $r_z \leftarrow r_{Q,z}[tid/2]$
2: $S[tid] \leftarrow 0$
3: for n = 1:N do
4:     $p_x \leftarrow p_{n,x}[n]$; $p_y \leftarrow p_{n,y}[n]$; $p_z \leftarrow p_{n,z}[n]$
5:     $r_n \leftarrow \sqrt{(r_x - p_x)^2 + (r_y - p_y)^2 + (r_z - p_z)^2}$
6:     if $tid$ is even then
7:         $S[tid] \leftarrow S[tid] + \frac{cos(kr_n)}{r_n}$
8:     else
9:         $S[tid] \leftarrow S[tid] + \frac{sin(kr_n)}{r_n}$
10: synchronize
11: if tid is even then
12:     return $\frac{\rho U_0 \lambda^2 f}{pps^2} \sqrt{S[tid]^2 + S[tid+1]^2}$
```

By separating the process, the two summations can be executed in parallel, resulting in a higher utilization of GPU cores. However, it introduces a synchronization point, which is enforced between corresponding pairs of threads.

Algorithm 1 details the kernel for the absolute pressure determination. The thread index is stored in variable *tid* and is unique to each thread. The total number of threads is double the number of elements of the discretized region of interest. The first step, described by lines 2 through 9, performs the summations. The vector $p[n]$ in line 5 stores the vector starting at the center of the circle and finishing at the square differential element. Then, after the synchronization, only half of the threads perform the last step, which yields the final absolute pressure.

4. EXPERIMENTS AND RESULTS

In order to evaluate the GPU based solution, a set of simulations were performed. These considered an ultrasonic transducer with circular radiant surface operating at different frequencies, ranging from 0.25 MHz to 5MHz, and emitting in a water medium. Other parameters considered were:

- speed of sound in water (c): $1500m/s$;
- water density (ρ): $1000kg/m^3$;
- normal displacement speed (U_0): $1m/s$;
- number of elements per wavelength (pps): 4.

The fixed value for the *pps* parameter ensure the same number of elements per wavelength. Hence, for higher frequencies, there are more elements and, consequently, the simulation has a higher computational load.

Two implementations were developed using the Visual Studio 2017 C++ compiler. The serial variant simply implements eq. (4) for each point in the domain. The second implementation consists of the proposed GPU parallel solution, which used the CUDA API. All tests were executed on a Intel Core i7 870 - 2.93 GHz computer with a memory of 16 GB and a GeForce GTX TITAN X GPU.

4.1 Results

Table 1 shows the execution times for the simulations performed using the CPU and GPU versions of the algorithm. The GFLOP (10^9 floating point operations) column was obtained through a simple procedure of counting all the operations, i.e., additions, multiplications, divisions, trigonometric functions and square roots. The speedup is the ratio between the serial and parallel execution times.

Table 1. Simulation results for the CPU and GPU implementations. *Freq.*: frequency in MHz. *N. elms*: total number of elements, *SU*: speedup

			Exec. time (s)		
Freq.	N. elms	GFLOP	CPU	GPU	SU
0.25	257	2.78E-04	0.000	0.001	0.30
0.50	1,740	1.04E-02	0.014	0.006	2.33
0.75	5,390	8.21E-02	0.104	0.013	8.00
1.00	11,963	3.34E-01	0.462	0.024	19.25
2.25	125,917	2.01E+01	24.208	0.585	41.38
3.50	459,317	1.82E+02	219.676	4.273	51.41
5.00	1,318,561	1.08E+03	1,319.840	24.369	54.16

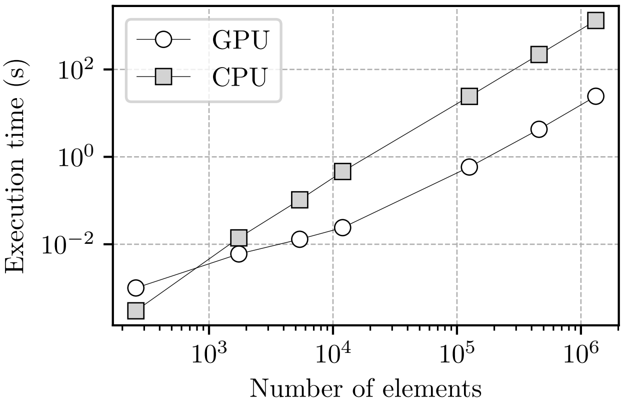


Fig. 2. Elapsed times for the two variants for acoustic field determination.

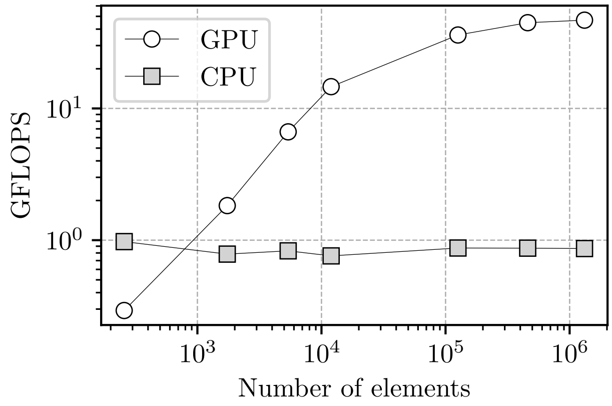
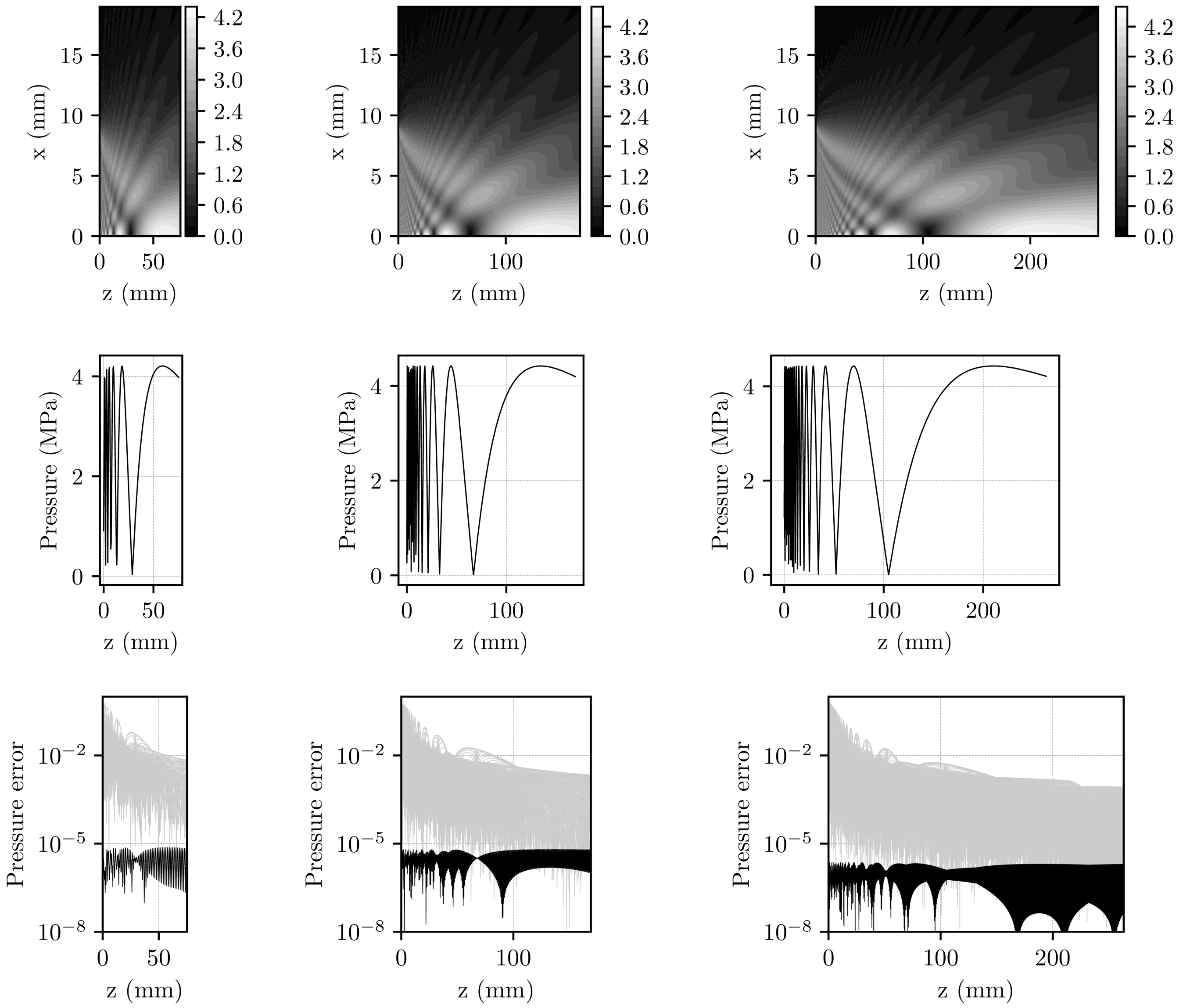


Fig. 3. GFLOPS for the two variants for acoustic field determination.

Figure 2 compares the execution times for both variants. It shows that the proposed implementation achieved better



(a) 1 MHz

(b) 2.25 MHz

(c) 3.5 MHz

Fig. 4. Top row: absolute pressure field (in MPa) for different frequencies. Middle row: axial pressure field. Bottom row: relative error over domain (multiple gray lines, all points with the same z are plotted with the same horizontal coordinate) and comparative error along the axis (black).

performance for all tests with at least 1740 elements. Also, as both curves grow exponentially, its impact in the speedup for higher frequency is substantial. By analysing the number of GFLOP per second, shown in Fig. 3, it is possible to determine that the GPU variant improves its performance as the number of elements increases. For a small number of elements (less than 1740), the host to GPU data transfer and setting operations in the GPU could take longer than the processing time in the CPU, bringing down performance. On the other hand, performance seems to stagnate around the 1.3 million elements, which corresponds to the 5 MHz frequency. This is consequence of the increasing in the data transfer operations into the GPU, spatially between the multiprocessors and global memory (see Table 1).

Finally, Fig. 4 shows the simulated acoustic fields (top row) and axial pressure (middle row) and relative error (bottom row) for 1.0, 2.25 and 3.5 MHz frequencies. The fields were calculated for a point slightly ahead of the near field limit. The well-known analytical expression for this distance ($z_n = D^2/4\lambda$) provides values of 66.7, 150 and 233 mm for 1.0, 2.25 and 3.5 MHz, respectively. These distances are close to the values observed in the axial pressure plots. Therefore, results are in agreement with the analytical solution reported in literature. Furthermore, the relative pressure error was computed between simulated acoustic fields and the King integral solution. The quantitative agreement between these numerical solutions gets better with the distance from the transducer surface to the propagation field. For reference, a comparative error analysis between the King integral and the analytical solution of the axial pressure field shows the agreement is quite good on the piston axis.

4.2 Discussion

The results showed that the use of the GPU-based implementation was advantageous in most of the experiments. The relative performance was influenced by the number of elements of the emission surface. Thus, it is possible to determine, from Fig. 2 that the minimum number of elements to achieve a positive speedup is approximately 1000.

Moreover, an estimate of the execution time for the GPU solution can be obtained from the speedup values in Table 1. As visual analysis of the plot from Fig. 3 indicates that the GPU was achieves maximum GFLOPS in the experiments, the speedup for larger frequencies can be estimated to be around 50. Finally, the results can extrapolated for different values of *pps*, by assuming that the number of elements is the main factor for the execution time of the algorithms.

5. CONCLUSION

A GPU-based solution for the determination of the acoustic field produced by a harmonically-excited circular transducer in a steady state was implemented. The simulations performed considered a range of frequencies between 0.25 and 5.0 MHz and the results indicated that the proposed solution outperformed the CPU implementation in all tests excluding one. The GPU variant was 50 times faster for the highest frequencies. The results can be employed to determine whether the GPU will contribute to the simulation and even to estimate the speedup. As a future work, the harmonic inputs can be combined to generate more generic functions.

ACKNOWLEDGEMENTS

The project is supported by Petrobras/ANP grant 5850. 0108871.18.9. A. Lemos D. is supported by CAPES. A. K. Sato and A. M. Silva Jr are supported by FUSP/Petrobras. M. S. G. Tsuzuki was partially supported by CNPq (proc. 305959/2016-6). J. C. Adamowski and F. Buiochi are partially supported by CNPq.

REFERENCES

Adamowski, J.C., Buiochi, F., Tsuzuki, M.S.G., Pérez, N., Camerini, C.S., and Patusco, C. (2013). Ultrasonic measurement of micrometric wall-thickness loss due to corrosion inside pipes. In *IEEE Int Ultrasonics Symp (IUS)*, 1881–1884.

Diaz, M.A., Solovchuk, M.A., and Sheu, T.W.H. (2018). High-performance multi-GPU solver for describing nonlinear acoustic waves in homogeneous thermoviscous media. *Comput Fluids*, 173, 195–205.

Franco, E.E., Andrade, M.A.B., Adamowski, J.C., and Buiochi, F. (2011). Acoustic Beam Modeling of Ultrasonic Transducers and Arrays Using the Impulse Response and the Discrete Representation Methods. *J Braz Soc Mech Sci*, XXXIII(4), 408–416.

Franco, E.E., Barrera, H.M., and Laín, S. (2015). 2d lid-driven cavity flow simulation using gpu-cuda with a high-order finite difference scheme. *J Braz Soc Mech Sci*, 37(4), 1329–1338.

Greenspan, M. (1979). Piston radiator: Some extensions of the theory. *J Acoust Soc Am*, 65(3), 608–621.

He, C. and Hay, A.E. (1993). Near-field characteristics of circular piston radiators with simple support. *J Acoust Soc Am*, 94(1), 554–561.

Johnson, J., Douze, M., and Jégou, H. (2019). Billion-scale similarity search with gpus. *IEEE T Big Data*, 1–1.

Kinsler, L.E., Frey, A.R., Coppens, A.B., and Sanders, J.V. (1999). *Fundamentals of Acoustics, 4th Edition*. John Wiley & Sons.

Kirkup, S.M. (1994). Computational solution of the acoustic field surrounding a baffled panel by the Rayleigh integral method. *Appl Math Model*, 18(7), 403–407.

Lebedev, G., Klimenko, H., Kachkovskiy, S., Konushin, V., Ryabkov, I., and Gromov, A. (2018). Application of artificial intelligence methods to recognize pathologies on medical images. *Procedia Comput Sci*, 126, 1171–1177.

Mehra, R., Raghuvanshi, N., Savioja, L., Lin, M.C., and Manocha, D. (2012). An efficient GPU-based time domain solver for the acoustic wave equation. *Appl Acoust*, 73(2), 83–94.

Nvidia (2018). Cuda c programming guide. Technical Report PG-02829-001˙v9.1, Nvidia.

Patrício, D.I. and Rieder, R. (2018). Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. *Comput Electron Agr*, 153, 69–81.

Piwakowski, B. and Delannoy, B. (1989). Method for computing spatial pulse response: Time-domain approach. *J Acoust Soc Am*, 86(6), 2422–2432.

San, J.E., Medina, M., Buiochi, F., and Adamowski, J.C. (2006). Numerical modeling of a circular piezoelectric ultrasonic transducer radiating in water. In *ABCM Symposium Serie in Mechatronics*, volume 2, 458–464.

Treeby, B.E. and Cox, B.T. (2010). k-Wave: MATLAB toolbox for the simulation and reconstruction of photoacoustic wave fields. *J Biomed Opt*, 15(2), 021314.

Tsiakas, K.T., Trompoukis, X.S., Asouti, V.G., and Giannakoglou, K.C. (2019). Shape optimization of wind turbine blades using the continuous adjoint method and volumetric NURBS on a GPU cluster. In E. Minisci, M. Vasile, J. Periaux, N.R. Gauger, K.C. Giannakoglou, and D. Quagliarella (eds.), *Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences*, 131–144. Springer International Publishing, Cham.

Wang, L., Spurzem, R., Aarseth, S., Nitadori, K., Berczik, P., Kouwenhoven, M.B.N., and Naab, T. (2015). nbody6++gpu: ready for the gravitational million-body problem. *Monthly Notices of the Royal Astronomical Society*, 450(4), 4070–4080.

Wang, X., Yin, H., Li, K., and Zhang, C.H. (2018). Implementing acoustic radiation force imaging on GPU using OpenCL. In *ICNC-FSKD 2017 - 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery*, 832–836.

Weight, J.P. (1984). Ultrasonic beam structures in fluid media. *J Acoust Soc Am*, 76(4), 1184–1191.