Efficient acoustic modelling of large acoustic spaces using finite difference methods

S Durbridge Bowers & Wilkins, West Sussex, UK

A Hill Department of Electronics, Computing & Mathematics, University of Derby, Derby, UK

# INTRODUCTION

* Introduce reason for doing study
  + Improvements in the flexibility, accuracy and performance of simulation tools could help towards making predictions, performing low-cost exploration (rapid prototyping) and system design workflows easier, faster and more intuitive. – WHAT
  + Time domain numerical methods used for performing acoustic simulation, could provide useful visual information, as well as reasonably accurate measurement data. – WHAT
* Lit Review – A bit more about time domain numerical methods for acoustic simulations
  + One of the early proponents of work on time domain numerical methods for acoustics was Bootledooren[1]; whose work involved porting the finite difference time domain (FDTD) and finite volume time domain (FVTD) methods from electromagnetic simulation, for use in low frequency acoustic simulation. – WHAT
  + This work has been followed on by many researchers such as Murphy[2], Bilbao[3] and Hamilton[4], to expand and improve potential use of these methods. – WHAT
  + In his thesis doctoral thesis[5], Hill presented a simple and effective implementation of the FDTD for low frequency modelling, that was the basis for the work presented below. – WHAT
  + Following key work such as that by Trefethen[6], a number of project have begun to explore the Fourier pseudo-spectral time domain method (PSTD), most notably the OpenPSTD project from Eindhoven University of Technology[7]. – WHAT
  + Caunce and Angus[8] recognised the limitations of implementing the FDTD method on general purpose graphical processing units (GPGPU) that can be used to improve the speed of FDTD solving, and introduced the potential improvements in execution speed by performing spectral differentiation on a GPGPU using the Fourier PSTD method. – WHAT
  + Meanwhile in the field of microcontroller development, Doerr[9] produced work on the spare finite difference time domain method for electromagnetic simulation. PIC microcontrollers are often modelled as vastly large electromagnetic simulations of networks of channels, that take large computational resources and a lot of time to simulate. The sparse finite difference time domain method presented by Doerr essentially presents a moving window method of reducing the size of the portion of the domain being solved for at any one time. - WHAT
  + Methods such as FDTD present benefits for low frequency simulation over other simulation methods such as ray based and direct calculation, as features such as the modal behaviour of the acoustic system being modelled is accounted for within the solving method. This is due to solving the acoustic wave equation in second order partial differential form; ray based methods assume planar radiation of sound waves, and ignore the radial propagation Eigen-modes of general acoustic systems. – WHAT/WHY
* Defined marker – regardless of approach it’s important to try and get a reduction in the time taken to do a simulation – MAIN GOAL
  + However large scale and high frequency simulations are difficult to perform using these methods, not only because of the large computational resources required to perform simulations; often the time required to perform these simulations is also severely limiting. – WHAT
  + If we get the speed up by some clever means (and ideally the memory restrictions down) the methods could be made more accessible for less specialist users i.e. PA and loudspeaker designers, students etc
* All above was setting the scene > there is this technology and it does this….
* Introduce the problem
  + Much of the literature has expressed how long it takes to do an FDTD simulation, and FEM and BEM can be faster, but isn’t always, and isn’t quite as flexible
  + An explanation of the problem in brief – Hard or impossible to model larger spaces on a normal PC with FDTD, is it worth exploring faster solvers
  + Introduce FDTD
  + Give some explanations of some practices in FDTD that may slow it down
  + Introduce PSTD and SFDTD and explain why this may speed things up
  + Explain the limitations of PSTD
  + Explain the limitations of SFDTD
  + Explain why those limitations are important in more detail
  + Figures backing up the problems
* Explain what this paper is all about – This paper is a cursory glance into a deep problem, that might be helped with some cunning future work
* Split out the content of the paper – the tests, results and such

This paper describes early research into the execution speed performance of time domain wave equation based acoustic modelling. The work was undertaken in Matlab, as part of an MSc project at the University of Derby.

# Experiment

* Introduce the experiment conditions: - WHERE (and why where)
  + Experiment done on a windows 10 PC with an I5 and 16GB Ram because I am poor
  + We Accept this was not ideal – but you gotta piss with the cock you got!
* A little bit more about the method – HOW of the thing measuring
  + Aim: to directly examine the time it took to get to a roughly similar solution
  + We used a series of domain sizes of the same shape and reflectance
  + Measuring points in a grid of so much
  + Centre of grid was in middle of the space
  + Domain size was dictated by how much memory I had access to
* Setup of the methods – HOW of the variable 1 domain setup
  + Three sets of domain setups required for each method, to keep stability and such
  + 1 – FDTD setup – dx dt etc
  + 2 – PSTD setup – dx dt etc
  + 3 – SFDTD setup – dx dt etc
  + WHY the setups? We want a reasonable behaviour, but it will never be true to real life – CAVEATS AND LIMITATIONS ARE IMPORTANT HERE
* Setup of the source and the stimulus – WHY?
  + Single omnidirectional source with tone-bursts & MLS
  + WHY that source? To find out if there is smear or other nasty behaviour, and stability too – FIGURES OF THIS DATA
  + Specifically, we would like to know the models are outputting similar things
  + Omnidirectional soft-source i.e. transparent, but not aligned with the impedance of the grid
* Defining the baseline – FDTD – HOW
  + The speed of an FDTD simulation was found, including the bottlenecks using profiler – WHY
  + Introduction of measuring software – CLIO was used
  + Introduction of stimulus – MLS
  + HOW – any other points?
  + Examined using spectral averaging? – WHAT
* Reviewing the results – WHAT & WHY
  + Spectral Shift – WHY
  + Data shows execution speed of X - WHAT
  + Profiler shows speeds of different parts- WHAT
  + Data size, array addressing etc - WHY
  + NOTE: NOISEFLOOR OF THE MEASUREMENT IS HIDDEN
  + NOTE: PLOT SCALE
* PSTD – HOW
  + A set of domains So big was used – WHAT – FIGURE 6 EXAMPLE OF THE CONFIGURATIONs
  + To see the scaling of the time taken, as I can only have domains of so big – WHY
  + Fc set by how large a domain could be fitted in memory – WHY
  + Would doing frequency domain differentiation be faster?– QUESTION?
  + Because we do simple contiguous addressing and arithmetic instead of differencing – WHY TO THE QUESTION
  + This is a different approach to differentiation, but might come with some drawbacks– WHY DO THIS TEST
* SFDTD – HOW
  + The next step was to try the SFDTD method to see if we can get speed improvements earlier in the simulation – WHY- FIGURE 7 EXAMPLE OF THE CONFIGURATION
  + Threshold value set to X – WHAT
  + Because it’s a minimum noise floor, and we spend loads of time below that not doing too much- but really its from dead reckoning – WHY
  + Same set of domains and fmax used for reasonable comparison – WHAT and WHY
  + The method would potentially be faster for the early reflections, because we chop down the size of the computation – WHAT & WHY
  + To give an idea of the behaviour of the strong wavefronts in a space and how these behave – FURTHER WHY
  + Because of the time limitations, the method wasn’t really optimised – WHAT
  + Explain that although the domain sizes and frequency used weren’t that big or high, it’s a starting point to go further– WHAT

# Results

* PSTD Results
* Reiterate the conditions of the experiment
  + X domain sizes and Y maximum frequency – WHAT
  + The figures below show the spectral output of the simulation at the measurement positions– WHAT
* Explanation of what the results show
  + It can be seen what the frequency responses in-front and behind with one piece of deck make little to no change In the audience area – WHAT
  + On and under the stage subwoofer positions there is a clear difference in frequency response with the deck in place, with a 15dB increase at 55Hz, and a collapse in the cardioid polar response when measured both on and under the stage – WHAT
  + With the subwoofer just in front of the stage, you get better on axis performance than the other two subwoofer positions
* Focus on the execution time profile
  + Its necessary to compare the frequency responses in more than on axis and off axis – WHAT
  + The 6 +/- 20degree measurement points were ignored because people aren’t normally there – WHY
  + The groups (onstage and audience) were averaged to give an idea of system behaviour in the audience and on the stage – WHAT & WHY
  + The averaged frequency responses with the single deck stage were taken from the no stage ones to give a deviation – WHAT & WHY
  + These responses are given in FIGURE 10 – WHAT FIGURES SHOWING DEVIATION
* Rounding off this part of the analysis of the speed of the PSTD simulations
  + This analysis provides ‘conclusive’ evidence that the best place to have a subwoofer around one piece of stage deck is in front of the deck – WHAT
  + This place exceeds or matches the front-to-back rejection ratio of the subwoofer with no deck – WHY
  + The under and on stage placements show reduction in the stage rejection in the subwoofers passband – WHAT
* SFDTD Results
* Introduce the SFDTD results analysis
  + An identical analysis to the small stage was done with PSTD – WHAT reiteration
  + The frequency responses are given in the below FIGURES
  + Threshold level of window was set to X – WHAT reiteration
* Focus on the execution time profile – what was slow
  + There are some similar and dissimilar trends – WHAT
  + The under stage location appears to be the worst choice in both cases, even though the large stage data was modelled – WHY/WHAT
  + So more investigation is needed – WHAT
  + Between 60 and 90Hz, there is less front-back rejection for the in-front of stage position than with no stage – WHAT
* Explanation of why SFDTD was faster at the beginning than the middle – BIG EXPLANATION OF WHY SOME UNEXPECTED VARIANCE
  + A possible explanation is the big stage is acoustically larger in the subwoofer passband, so most frequencies interact with the stage - WHAT
  + A wall was just behind the stage, so a strong reflection may have interacted with the measurements – WHY
  + The propagation distance of the reflection was 9.7 meters for the first driver of the subwoofer, a wavelength relative to 17.68Hz - WHAT
  + Odd integer multiples of this arrive at the sub 180degrees out of phase with the direct output of the sub, causing cancellation in front of the drive unit – WHY
  + Key frequencies of this odd order multiples are 53Hz and 89Hz – WHAT
  + The propagation distance of the reflection for the second driver is 8.7 meters, equating to integer multiples of 59 and 99Hz – WHAT
  + The significance of this loss of stage rejection requires further work – WHAT
* The execution speed comparison
  + Finally
  + Mean front-back SPL rejection over two frequency ranges (38 – 110Hz) and (20-300Hz) – WHAT
  + Wider range to account for stage and room resonances –WHY
  + Is given in table 1 & 2 – WHAT
  + The data in those tables give a clear and concise summary – WHAT
  + Placing a cardioid subwoofer on top or under a stage, regardless of stage size, will reduce the front-back rejection ratio a lot! – WHAT
  + Placing the subwoofer in front of the stage will allow the rejection to be maintained – THE BIG OUTCOME OF THE STUDY – THE WHAT

# CONCLUSION & Further Work

* Review of the results and final outcome
  + The results in the paper aid further understanding of the problem – the effect of the stage on the polar response – WHAT
  + If a sub is under or on top, the speaker won’t have the same great directivity – WHAT
  + The result is high SPLS on stage which is no good – WHAT
  + Placing the sub in front of the stage is the best for maintaining the polar response – WHAT
  + The results here and the results from earlier work show that its best to place the sub in front of stage – WHAT BIG WHAT
* Caveats
  + This analysis is meaningless if the subwoofers are flown above the stage – Counter point
  + If you do this, do beam steering so that there is less noise on the stage
* Further work
  + While this work shows some good evidence, more work needs to be done to fully know what is going on – WHAT
  + These recommendations for further work – WHAT
    - Repeat experiment in anechoic space with full stage
    - Repeat experiment in a large scale live event
    - Examine how multi-unit cardioid subwoofer arrays interact with each-other in different shape arrays with different stage positions
    - Investigate effects the stage has on transient response
* Fourth paragraph
  + Although more work needed to understand, it is clear that you should think about where to place ground based subs at events, especially when directivity is important. – WHAT WIDER
  + Most commercially available software omits any stage effects, so it’s essential to know what the stage does to sub cardioid performance, so not to mess up and to get best response shape – WHAT

# References

[1] J. De Poorter and D. Botteldooren, “Acoustical finite-difference time-domain simulations of subwavelength geometries,” *J. Acoust. Soc. Am.*, vol. 104, no. 3, pp. 1171–1177, 1998.

[2] D. T. Murphy, A. Southern, and L. Savioja, “Source excitation strategies for obtaining impulse responses in finite difference time domain room acoustics simulation,” *Appl. Acoust.*, vol. 82, pp. 6–14, 2014.

[3] S. Bilbao, “Optimized FDTD schemes for 3-D acoustic wave propagation,” *IEEE Trans. Audio, Speech Lang. Process.*, vol. 20, no. 5, pp. 1658–1663, 2012.

[4] B. Hamilton and S. Bilbao, “Fourth-Order and Optimised Finite Difference Schemes for the 2-D Wave Equation,” *Research.Ed.Ac.Uk*, vol. 2, no. 2, pp. 1–8, 2013.

[5] A. J. Hill, “Analysis , Modeling and Wide-Area Spatiotemporal Control of Low-Frequency Sound Reproduction,” University of Essex, 2012.

[6] L. N. Trefethen, “Spectral Methods in Matlab,” *Lloydia Cincinnati*, vol. 10, p. 184, 2000.

[7] M. Hornikx, T. Krijnen, and L. Van Harten, “OpenPSTD: The open source pseudospectral time-domain method for acoustic propagation,” *Comput. Phys. Commun.*, vol. 203, pp. 298–308, 2016.

[8] J. A. S. Angus and A. Caunce, “A GPGPU Approach to Improved Acoustic Finite Difference Time Domain Calculations,” *128th Audio Eng. Soc. Conv.*, 2010.

[9] C. Doerr, “SPARSE FINITE-DIFFERENCE TIME DOMAIN SIMULATION,” US 2014/0365188 A1, 2014.