

1 Introduction

Interest in acoustic modelling continues to increase due to external factors such as a renewed focus in global standards of distributed loudspeaker system design, as well as trends in consumer technology such as the video games and entertainment applications. This brings acoustic simulation to the forefront for areas such as public address and safety(Public Address Voice Alarm(PAVA))system design, as well as less safety driven interests such as immersive virtual reality experiences of synthesized spaces for entertainment. The benefits of improved acoustic modelling methods allow key stakeholders to leverage high quality results and create better quality products. Time is an ever more strict constraint in many commercial development applications, and an inherent issue with high accuracy modelling is a significant increase in time spent simulating, as well as computational resources and specialist knowledge.

1.1 Problem Definition

Real time acoustic modelling would be of significant benefit to many fields, engineers could make design changes and see results 'on the fly', and entertainment users could have more realistic experiences. This should be possible for an arbitrary number of sources and receivers, in a proportionally large environment at high quality. Is it possible to further reduce computation time for simulations of large acoustic problems? There are two 'branches' of computation solution that should be considered: the direct solution i.e. direct outputs or audio samples from the simulation, and indirect solutions i.e. a system impulse response that may be convolved with mixed source signals in order to create an auralization of the system.//



Fig. 1. A visualisation of virtual sound sources [?]

2 Acoustic Modelling

2.1 Applications

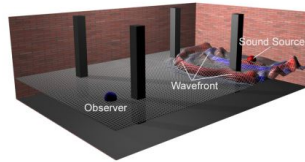
Acoustic modelling has a broad range of applications from the very large to the very small, and there are a number of key methods used to model different problem types. Problems of interest to this study are the acoustically large i.e. problems that are significantly large compared to a high proportion of the wavelengths of interest. These problems could include multiple sources and receivers in spaces such as stadiums, cathedrals, shopping centres, caverns, cityscapes, mountain ranges etc.

For problems of this nature, ray based modelling methods are commonly used. Ray based methods have a significant number of compromises, such as they do not accommodate for modal or atmospheric behaviour. Further to this when considering a large number of sound sources and receivers in a large simulation, some of the benefits of ray based methods may be outweighed with computational cost. Wave based modelling methods have a similar set of benefits and drawbacks, that have as yet reduced use to high detail simulation of smaller problems. As higher computational power becomes more widely available, wave based methods may become more practical to use for problems with varying numbers of sources and receivers (particularly moving sources and receivers) e.g. video games, building acoustic demonstrations, personalised audio for viewing moving pictures, pilot training simulation etc. Below is a brief discussion of ray and wave based methods, with a deeper focus on time domain wave methods.

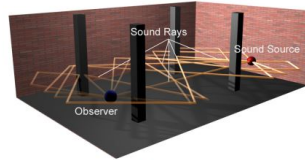
2.2 Ray Methods

Ray based modelling methods, also regarded as geometric methods, are those that rely on the concept of ray casting in order to compute the impulse response of an acoustical system¹. In these methods rays are cast between sources and receivers, with the time of flight and boundary interactions(reflections) accounted for [?]. As such this method relies on the assumption that the acoustic wavelength being modelled is significantly smaller than the smallest potential surface segment. Varieties of geometric methods include ray tracing, beam tracing and the image source model. These methods does not account for modal effects at low frequency, and may require a significant number of rays for each source and receiver, in order for any result to be relevant for large problems. Further to this, a new model has to be computed every time a source and receiver moves to a different location or is 'aimed' in a different direction within the geometry. However, ray tracing is a relatively mature and well supported method in graphical rendering of environments, and may benefit from continual support and optimisation for faster performance.

¹In the context of this report, an acoustical system is the combination of sound source, sound receiver and surrounding environment



(a) Wave-based Approach.



(b) Ray-based Approach.

Figure 1: Acoustic Simulation Techniques.

Fig. 2. A visualisation of ray tracing [?]

2.3 Wave Methods

Wave based modelling methods comprise of the discretization of a problem geometry into rational segments, following which a partial differential equation is solved across the components or 'elements' of that geometry. The partial differential equations used in the context of large problems are generally linearised Navier-Stokes equations for waves in homogeneous media. Some methods will perform in the time domain i.e. iterative direct solutions, and some in the frequency domain such as FEM or the boundary element method (BEM). The benefit of 'element' methods such as FEM, BEM Discontinuous Galerkin, is that a significant amount of supporting research has been done to improve this method of formulation. This includes meshing using arbitrarily sized elements, applying and combining different physics problems, and has generally been improved for relatively fast and efficient solving. In acoustics, these methods still involve solving a domain that is discretized such that the smallest element is a sixth the size of the shortest wavelength. Further more frequency domain calculations require solutions for every frequency of interest, and can be much slower than required for real-time computation. Time domain methods for direct solutions can however be optimised for real time direct solutions. Three time domain methods are evaluated in the next section.

3 Time Domain Methods

3.1 FDTD

The finite-difference time domain(FDTD) method as described by Yee, involves discretizing the domain into 3 separate components for a 2 dimensional problem. These components for an acoustic case are velocity vectors in each dimension, and a uniform pressure grid. In the basic 7 point stencil the difference between the pressure and surrounding velocity gradients at a half step in time, are evaluated to determine the pressure at that location for the next half step in time. This method is a second order central difference in space and first order forward difference in time(calculating the future, based on the past and present). For acceptable stability fulfilling the Courant condition, upwards of 6 cells per smallest wavelength is required per simulation. Generally, FDTD 'stencils' requires a purely rectilinear discretization of the problem geometry, though octahedral and other schemes have been used in the similar finite-volume time domain method. Though relatively simple and scalable, basic FDTD methods require all local calculations to be undertaken, and as such are more prone to error as well as requiring a large number of memory accesses. However, as the stencil equations is identical within the problem geometry (excluding the boundaries), it is conceptually highly parallelizable i.e. the same basic math operations are used to evaluate a large data set in fast succession.

3.2 S-FDTD

The sparse finite difference time domain method is simply an extra step added to the FDTD method, in which the power distribution of the grid is evaluated and only cells within a rational distance of those with an appropriate amount of energy are computed. This method is appropriate for electromagnetic simulation as the waves are transverse and can propagate through contiguous media (or no media) more simply than the acoustic case. As acoustic waves are longitudinal the SFDTD method may be applied to pulses such as Gaussian functions in order to still remain relevant, in this the travel and decay of wavefronts is accounted for in the calculation, and multiple sources and receivers can be computed for simultaneously. Some modal behaviour may not be incorporated in this method, but this must be confirmed with experimentation.

3.3 PSTD

The pseudospectral time-domain method (PSTD) is a happy medium of high accuracy frequency domain differentiation, with simplicity and efficiency of time domain solving. In this method, rows and columns are transformed into the frequency domain, and are differentiated all together by being multiplied by a differentiating impulse response. The resulting grid is then evaluated with the appropriate coefficients and an attenuating perfectly-match layer is applied as an absorbing boundary condition.

3.4 System Speed

In order to obtain 'real time' performance in a non real time system such as a generic operating system, the modelling method must be able to receive, compute and return buffer-loads of audio samples in fast succession with minimal latency i.e. having a system that performs optimally fast to allow for a minimal buffer size. Direct solutions may be beneficial for very large problems with a large or fluctuating numbers of sources and receivers, as no extra computations will be necessary so the time distributed by the kernel can be used optimally for the solution processing. An indirect solution may still be beneficial to have, as a system of indirect solutions can be cross-faded between. This means that if the computation process is running out of time the previous solution can still be used, although this solution is effectively incorrect there will be less chance of returning a buffer without the correct number of results.

In Windows distributions before 10, the stock buffer size was 16mS which may equate to a 768 sample buffer. However, much of the buffer size is dependent on hardware drivers. As of Windows 10, buffer sizes default to 10mS i.e. 480 samples large. However, as windows is a non-real time multi-tasking operating system, there is no trivial answer to how fast the process must perform beyond 'as fast as possible' with the largest problem size.

4 Conclusion

5 References