

ADEAS: A Distributed Environment for Acoustic Simulation

V. V. Arni & T. L. Huntsberger
Intelligent Systems Laboratory
Department of Computer Science
University of South Carolina
Columbia, SC 29208
terry@cs.scarolina.edu

B. A. Huntsberger
Parallel Supercomputer Initiative
College of Science and Mathematics
University of South Carolina
Columbia, SC 29208
bev@grover.csd.scarolina.edu

Abstract

Acoustic properties of environments are difficult to model, due to their dependence not only on geometry, but also on the internal properties of materials (density, etc.). An acoustically accurate model that can be quantitatively validated is extremely useful for the evaluation of improved enclosures, silencers, acoustically absorbent materials, and human factors studies. In this paper, we present a new parallel algorithm that generates full 3-D simulations of speech intelligibility and sound pressure levels which include sound diffraction, sophisticated external noise and surface absorption models. The simulation environment interface is decoupled from the algorithm using a C++ framework running under X11R5 and Motif. The sequential version of this algorithm produced sound pressure levels that agreed with measured values on the STS-40 Space Shuttle mission to within 4%. We also present the results of some experimental studies performed using ParaSoft Express on the 56-node Intel Paragon at the University of South Carolina.

1 Introduction

Acoustic analysis of enclosures can be broadly broken into two approaches, the statistical and geometrical. The most well known representatives of these two methods are ray tracing for statistical, and the source image model for geometrical analysis. The ray tracing method resembles the optical ray tracing analysis used in computer graphics. A discrete number of sound rays are shot into the environment from each source. An attempt is made to simulate the spherical propagation of sound waves. The source image model uses sound ray paths calculated between all possible combinations of reflecting surfaces in the environment. As such, it is closer to the radiosity approach used for realistic im-

age generation in the computer graphics field. Since ray paths are known exactly, sound absorption by the surfaces of the enclosure can be calculated based on physical properties.

There are a number of problems with the commonly used ray tracing approach to acoustic modeling in cluttered, confined non-homogeneous areas such as those found in most environments. Among these are:

- There are no rules for deciding the number of initial rays for the simulation, which leads to the possible neglect of valid ray paths and different results being obtained for the same environment [1]. There is no theoretical method for knowing when valid paths have been neglected [2].
- The discrete sampling approach is not as valid for medium and high frequency sound waves, since the wavelengths in these cases are about the same size as objects in the environment (from 33cm for a 1100Hz sound wave to 33mm for a 11kHz sound wave). There are considerable diffraction effects around edges for such sound waves [3].
- Loss of sound energy to surfaces in the environment is difficult to calculate exactly since the ray tracing approach gives rise to a diffuse approximation to the actual sound wave [12]. Long term acoustic vibrational effects which are dependent on these types of measurements would not be very accurate in this model. In addition, violations of the diffuse sound assumption are present in the case of localized intermittent sound sources, such as those found in a large number of environments.

Being a statistical method, the ray tracing approach works well for large rooms such as concert halls, as was

demonstrated in an experimental study conducted by Gimenez and Marin [5] and a visualization study by Stettner and Greenberg [20]. However, the possible inaccuracies in the model due to the points mentioned above make it a less optimal technique for acoustic simulations.

The source image model approach addresses all of these potential shortcomings of the ray tracing method. Multiple sources and receivers can be simulated in a straightforward manner. Since ray paths are calculated for every combination of surfaces, no valid ray paths will be excluded with such a procedure and results for a simulation are reproducible, as opposed to the situation with the ray tracing approach. In addition, frequency dependence of diffracted sound waves can be explicitly included in the model since the paths can be examined for object corner collisions [14].

The generalized algorithm of Borish extends to any arbitrary polyhedral room [2], and computational concerns of storage and time have been addressed by Lee and Lee [13]. A recent study by Hammad indicated the utility of the source image model technique for calculation of sound pressure levels, and clarity or definition in an enclosed environment [7].

Despite the improved algorithm of Lee and Lee [13], the time complexity of the sequential algorithm is of $O(N^o)$ time where N is the number of reflecting surfaces and o is the order of reflection. For example, a million paths would have to be evaluated for a second order study of one thousand reflecting surfaces. This operation would have to be performed for each grid point within the three dimensional sound enclosure. The sequential version of such an algorithm would be computationally bound [9].

This paper presents a new parallel algorithm called ADEAS (A Distributed Environment for Acoustic Simulation) which uses the source image model to generate full 3-D simulations of speech intelligibility and sound pressure levels. These simulations include sound diffraction, sophisticated external noise and surface absorption models. The application portion of the system was written using C and the ParaSoft Express [17] language. The user interface portion has been decoupled from the application using a C++ graphical design. This approach isolates the user from the details of the application that is running on the distributed memory platform. The next section describes the source image model. This is followed by a discussion of the ADEAS design. Finally, the results of some experimental studies are presented.

2 Source image model

Reflections of sound waves from the surfaces of an enclosure can be derived using images of the sound sources in an environment. Each sound source is mirror reflected through a surface and the intersection and reflection of the sound ray is given using this source image. This process is shown in Figure 1, where S_0 is the original sound source position and S_1 is the mirrored position.

The previous work for arbitrary bounding polyhedral surfaces for enclosures were limited due to large

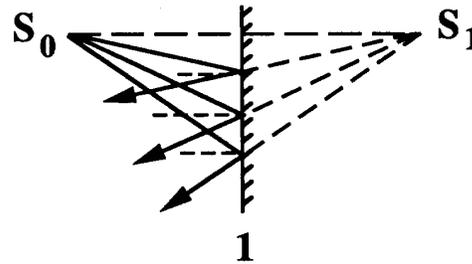


Figure 1: Source image model for reflection from a surface.

storage and computer time costs [2]. A new analysis by Lee and Lee [13] has substantially cut these costs through the development of an efficient algorithm. This new algorithm is based on the geometrical transformations necessary for multiple mirror reflections being expressed as matrix transformations.

Each of the mirror transformations is actually a change of coordinate system. Since the reflection from a boundary of the enclosure is a mirror transformation, a single reflection is one mirror transform, two reflections from different surfaces would be two mirror transformations and so on. For example, if the mirror reflection across surface 1 for the source S_0 is desired, this transformation can be expressed as

$$S_1 = S_0 [T_1], \quad (1)$$

where $[T_1]$ is the transformation matrix. For a sound path that involves reflections from multiple surfaces, the sound source must be mirrored through each surface. The source image position for a path which includes surfaces 1 and 2 would be given by

$$S_{12} = S_0 [T_2] [T_1] = S_0 [T_{12}]. \quad (2)$$

In general for the k th order image source,

$$S_{i_1 i_2 i_3 \dots i_k} = S_0 [T_{i_1 i_2 i_3 \dots i_k}], \quad (3)$$

where k is the number of surfaces in the path.

Each transformation will require rotational and translational components. Lee and Lee were able to derive a nice compact form for this transformation matrix [13]. Since any number of reflections can be generated from the multiplication of the appropriate matrices, the transformation need only be calculated once for all surfaces of the enclosure. This transformation basis is used for sound path generation in ADEAS.

All of the rays generated through the use of the matrix transformations are not valid rays. The transforms were done using the infinite plane assumption for the mirror plane. However, some intersection points would be generated that are not within the boundary of the reflecting wall. A simple cross-product test can be used to determine whether the ray

actually intersects the wall [2]. If an invalid portion of a ray path is found, then that total ray path is rejected. The other type of invalid ray is that due to obstructions in the environment. In this case a ray intersects a surface which intrudes into the enclosure, and once again any such intersection invalidates the entire ray path.

2.1 Absorption of sound

As a sound wave propagates through an environment, there are two main sources for loss of energy: loss to the air and to the boundaries of the enclosure during reflections. The energy loss to the air is dependent on the path length and is given by

$$E = E_0 e^{-m_a l}, \quad (4)$$

where E_0 is the original energy of the ray, l is the path length and m_a is the coefficient of attenuation of sound energy in air [15]. This coefficient has a strong dependence on the frequency of the sound wave and the humidity of the air in the environment. ADEAS includes this dependency.

Every time a sound ray hits a surface and is reflected, a portion of its energy is lost to the wall which may re-emit it as vibrational energy. The reflected energy is given as

$$E = E_0 (1 - \alpha), \quad (5)$$

where α is the sound absorption coefficient of the wall. The energy for a sound wave undergoing multiple reflections E_{ref_i} is given by

$$E_{ref_i} = E_0 e^{-m_a \sum_k l_k} \prod_k (1 - \alpha_k), \quad (6)$$

where k is the number of reflections for each path i . The total energy reaching a receiver must include the direct path between source and receiver, where the only loss is to the air. If l_{SR} is the length of this path, then the total energy is

$$E_{TOT} = E_0 e^{-m_a l_{SR}} + \sum_i E_{ref_i}. \quad (7)$$

The total energy reaching a receiver is not a useful measure unless the temporal properties, such as reverberation time of the enclosure, are also included.

2.2 Speech intelligibility

One of the major concerns in most acoustic simulations is the effect of ambient noise levels on the intelligibility of speech. There are two items relevant to this problem, the sound-pressure level of the interfering noise and the necessary speaker output to overcome this level for intelligible speech. Some experimental work by Haas has indicated that the speech tempo, intensity of the signal, tone color and acoustic characteristics of the enclosure all influence the intelligibility of speech [6]. In particular, reduction of tempo from

7.4 to 3.5 syllables per second, or an increase in intensity from 0dB to 6dB, or a suppression of high or low frequencies all increase the intelligibility of speech.

A recently introduced model using the modulation transfer function of an enclosure includes all of these factors in addition to ambient noise in the environment [8]. An enclosure can effect the intelligibility of speech through the indirect reflections from the boundaries and through the ambient noise present in the environment. The basic premise behind the model is that the sinusoidal components of the sound envelope are preserved throughout the signal degradation process. A measure of this component preservation as a sound wave travels from a source to a receiver is the modulation transfer function (MTF). The speech transmission index (STI), which has been shown to correlate extremely well with the commonly used articulation index (AI) [4], can be directly calculated from the MTF [8]. The relationship between the MTF and the STI is derived from an apparent signal to noise (S/N) ratio for the system.

If the source signal has a modulation frequency f (syllables per second), then the MTF for this system is given as

$$MTF'(f) = MTF(f) \frac{\sum_n I_s(\theta) \frac{a_n}{r_n^2}}{\sum_n I_s(\theta) \frac{a_n}{r_n^2} + I_n}, \quad (8)$$

where the MTF for the system without noise is

$$MTF(f) = \frac{\left| \sum_n \frac{a_n \exp(-2\pi j f r_n/c)}{r_n^2} \right|}{\sum_n \frac{a_n}{r_n^2}}, \quad (9)$$

and a_n is the attenuation from absorption by surfaces for each path n , r_n is the length of the path, c is the velocity of sound, $I_s(\theta)$ is derived from sound source distribution information similar to the goniometric diagrams used in optical image generation, and I_n is the intensity of noise at the receiver position.

From this expression, the STI can be calculated directly as

$$STI(f) = \frac{S/N_{app}(f) + 15}{30}, \quad (10)$$

where

$$S/N_{app}(f) = 10 \log \left(\frac{MTF'(f)}{1 - MTF'(f)} \right) \quad (11)$$

is the modulation frequency specific apparent S/N ratio. $S/N_{app}(f)$ has been clipped when exceeding the range ± 15 dB for normalization purposes. An example of the STI for a slice through a sample rectangular enclosure is shown in Figure 2. There are two directional noise sources at either end of the room, and the speaking source against the lower wall is directed toward the upper right corner of the room. Gray scale is being used to indicate the value of the STI, which ranges from dark (0.0) to white (1.0). Values above

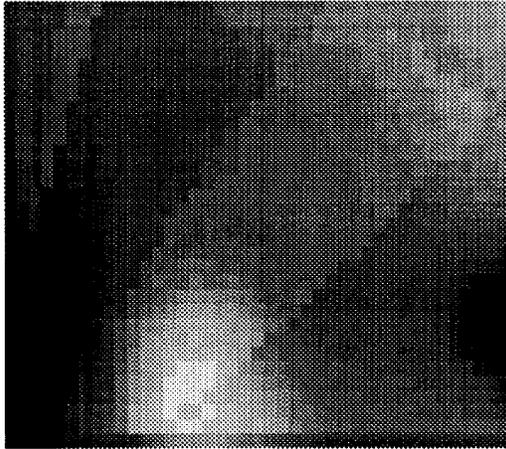


Figure 2: STI for sample enclosure.

0.6 indicate good speech intelligibility. The room has strongly accentuated regions where the STI is still high far away from the source, such as in the upper left corner.

3 ADEAS design

Jhaveri developed a uniprocessor model for acoustic simulation [9], based on Lee and Lee's source image technique and took into account the absorption of sound and reverberation [13]. This model included the effect of diffraction from edges. Even though this model was successful in studying different environments, it was found to be very slow as the number of reflecting surfaces in the enclosure and the order of reflection increased, thus prompting a distributed approach.

ADEAS has been designed in a highly object oriented, modular and structured fashion. ADEAS consists of two major parts, the user interface and the application. These two parts are joined by a C++ class object called *AppInt* which acts as an interface between the user interface classes and the application program. The user interface has been implemented in C++ and Motif Application Framework using object oriented techniques. The functions of these classes are explained in the next section. The application part of the program has been implemented in C. This paradigm was chosen in order to isolate the user from the details of the distributed algorithm.

The user interface portion of ADEAS consists of C++ classes developed over the Motif Application Framework (MotifApp) classes. The framework consists of a set of base classes which defines the basic structure common to many applications. It also defines the flow of control of the whole interface. Additional classes, peculiar to this project, were developed

over the base classes. These classes handle the parameter setting and visualization aspects of the user interface interaction with the distributed application program.

The model classes are used for the setting of graphics options like viewpoint and enclosure volume visualization. The utility classes directly interact with the application program running in the distributed environment. The *AppInt* class is a type of utility class and except for this class, no other class has direct contact with the application part of the program. The command classes are used for passing information to the *AppInt* class to forward to the application program. Finally, the button interface classes are the direct link to the visual display portion of ADEAS.

The application program is implemented as a host program which is responsible for managing the user interaction, disk file I/O and initial data transfers to the nodes; and a node program duplicated on each processor. For a given enclosure, the host passes the same number of reflecting surfaces to each node, equal to the total number of surfaces in the enclosure divided by the total number of nodes in the system. This surface data includes shape and position information, and absorption characteristics for each surface. The host also broadcasts data common to all the nodes, including the locations of the source and the receiver, and environment parameters such as air temperature and humidity.

Each node receives its own unique set of surfaces from the host and independently calculates the transform matrices for those surfaces. In this way, the transform matrix calculations are performed in parallel. The nodes then broadcast each set of transform matrices and coefficients to all the other nodes, thereby receiving an additional set from each of the other nodes. Through this process, all the nodes assemble an identical set of transform matrices, which is the complete set for all the original surfaces. For concave enclosures, the non-facing surfaces list is generated next, in an analogous fashion to the transform processing; with parallel computation of subsets, followed by a broadcast of the results from each node to all other nodes.

Next, each node begins the independent, parallel generation of all paths whose first point of reflection is on any of the surfaces contained in that node's unique set of surfaces. With the complete transform matrix data available, each node can generate all the valid paths for its own set of surfaces. The product of transform matrices calculated at each order of reflection can be reused in the product calculations at the next higher order. Once the ray paths are available, the sound pressure levels and STI values are independently calculated and sent to the user interface for display and interpretation.

The most computationally intensive part of the acoustic simulation is the generation of the sound wave paths from the source to the receiver. For an order of reflection o and number of reflecting surfaces N , the total combination of transformation matrices that are to be considered for the generation of sound paths

are [13]

$$SR_{tot} = N + N(N - 1) + \dots + N(N - 1)^{o-1} \quad (12)$$

The total number of sound ray paths SR_{tot} increases in geometric progression with the increase in the number of reflecting surfaces and order of reflection. In the above expression, N gives all first order paths, $N(N - 1)$ gives all second order paths, and so on. The calculation of the transformation matrix of each surface is in itself computationally intensive.

4 Experimental studies

Our algorithm is based on a complete implementation of this model, originally developed by Arni [1]. First, we adapted the matrix transform and path generation portions of the original implementation from the Reactive Kernel [18, 19] to Network Express. The Express version was developed on various distributed workstations and then moved to the 56-node Intel Paragon at the University of South Carolina.

Procs	512 surfaces		2048 surfaces	
	Order 1	Order 2	Order 1	Order 2
1	0.119	12.477	0.475	197.007
2	0.063	6.256	0.241	98.560
4	0.032	3.139	0.122	49.304
8	0.017	1.583	0.062	24.706
16	0.009	0.815	0.032	12.397
32	0.005	0.433	0.017	6.278
Paths	102	19	199	9

Table 1: Calculation times (in seconds) and number of valid paths.

In this paper, we report only the times and efficiencies for the transform matrix and path generation processes running on the Paragon. As was stated in the previous section, the majority of the calculation time is taken in these two steps. The times reported in Table 1 are from convex input datasets only, since the current Express version of the algorithm is limited to these types. The two datasets reported here are spherical enclosures, consisting of 512 and 2048 reflecting surfaces, respectively.

Table 1 gives the calculation times in seconds for first and second orders of reflection, for both datasets, for a varying number of processors. As shown, with an increase in the number of surfaces, the overall times increase rapidly, especially for the order 2 problems. However, holding the dataset size constant and increasing the order from first to second does not result in the expected geometric increase in time. The expected calculation time for each higher order is given by a geometric progression in the number of reflecting surfaces, as shown in equation (12). In fact, for both datasets, the second order runs require less than 21% of the expected calculation times based on the first order times. This is due in part to the fact that large

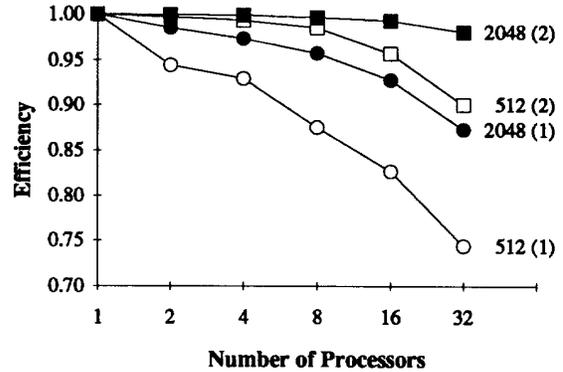


Figure 3: Efficiencies for ray paths of first and second order reflections.

numbers of possible paths are determined to be not valid at early stages. For example, the maximum possible number of paths for the 2048 (order 2) dataset is on the order of 4 million. Table 1 shows the numbers of valid paths for each case, and that only 9 valid paths exist for that example.

Figure 3 shows the associated efficiencies, labeled by size of dataset with order of reflection in parenthesis. As can be seen, the scaling over the number of nodes improves with higher order path generation, as well as with larger datasets. There are simply not enough first order calculations on the smaller enclosure (512 surfaces), to maintain efficient use of larger numbers of processors. The distributed calculations are relatively insignificant with respect to the constant setup time required by every node. As discussed previously, second order calculations have a much higher level of complexity. Figure 3 demonstrates that this increased complexity is the dominant factor leading to the improvement in efficiency. Any degradation in efficiency caused by adding the node-to-node broadcast of transform matrices (which is not needed for order 1 runs) is relatively insignificant.

5 Discussion

We have presented a new distributed memory algorithm for the simulation of acoustic enclosures. This algorithm generates full 3-D simulations of speech intelligibility and sound pressure levels which include sound diffraction, sophisticated external noise and surface absorption models. An X-Windows interface with interactive parameter settings serves as the front-end to this algorithm, thus isolating the researcher from the details of the underlying distributed memory implementation. The experimental studies on the Intel Paragon at the University of South Carolina using the ParaSoft Express language indicate that the algorithm has good scaling properties over the number of reflecting surfaces, the order of reflection and the number of

nodes. The efficiency for second order calculations on the larger dataset was still 98% at 32 nodes.

We will be investigating larger datasets, including the Space Shuttle crew bay model from NASA Johnson which is highly concave and has over 10,000 reflecting surfaces. In addition, quantitative studies are underway to further validate the simulated sound pressure levels and speech intelligibility results.

Acknowledgements

We wish to thank Ken Sallenger and Pat Fay for insuring the necessary hours of Paragon access time. We also thank Ken for his after-hours ministrations to the SPARCstation network.

References

- [1] V. V. Arni, "ADEAS: A distributed environment for acoustic simulation," M.S. Thesis, Department of Computer Science, University of South Carolina, 1994.
- [2] J. Borish, "Extension of the image model to arbitrary polyhedra," *J. Acoust. Soc. Am.*, Vol. 75, No. 6, June 1984, pp. 1827-1836.
- [3] J. J. Bowman, T. B. A. Senior and P. L. E. Uslengi, *Electromagnetic and Acoustic Scattering by Simple Shapes*, North-Holland, 1976.
- [4] N. R. French and J. C. Steinberg, "Factors governing the intelligibility of speech," *J. Acoust. Soc. Am.*, Vol. 19, 1947, p. 90.
- [5] A. Gimenez and A. Marin, "Model of energy calculation for acoustic analysis of closed rooms," *Applied Acoustics*, Vol. 26, 1989, pp. 99-111.
- [6] H. H. Haas, "Über die Einfluss eines Einfachechos auf die Hörbarkeit von Sprache," *Acustica*, Vol. 49, 1951.
- [7] R. N. S. Hammad, "Simulation of noise distribution in rectangular rooms by means of computer modelling techniques," *Applied Acoustics*, Vol. 24, 1988, pp. 211-228.
- [8] T. Houtgast, H. J. M. Steeneken and R. Plomp, "Predicting speech intelligibility in rooms from the modulation transfer function. I. General room acoustics," *Acustica*, Vol. 46, 1980, pp. 60-72.
- [9] J. Jhaveri, "Simulation of Speech Intelligibility", M.S. Thesis, Department of Computer Science, University of South Carolina, 1992.
- [10] V. O. Knudsen and C. M. Harris, *Acoustical Designing in Architecture*, John Wiley and Sons, 1968, p. 155.
- [11] A. Krokstad, S. Strom and S. Sorsdal, "Fifteen years experience with computerized ray tracing," *Applied Acoustics*, Vol. 16, 1983, pp. 291-312.
- [12] A. Kulowski, "Algorithmic representation of the ray tracing technique," *Applied Acoustics*, Vol. 18, 1985, pp. 449-469.
- [13] H. Lee and B.-H. Lee, "An efficient algorithm for the image model technique," *Applied Acoustics*, Vol. 24, 1988, pp. 87-115.
- [14] Z. Maekawa and S. Osaki, "A simple chart for the estimation of the attenuation by a wedge diffraction," *Applied Acoustics*, Vol. 18, 1985, pp. 355-368.
- [15] V. S. Mankovsky, *Acoustics of Studios and Auditoria*, Focal Press Limited, Great Britain, 1971.
- [16] M. L. Mehta and K. A. Mulholland, "Effect of non-uniform distribution of absorption on reverberation time," *J. of Sound and Vibration*, Vol. 46, No. 2, 1976, pp. 209-224.
- [17] Express Reference Manual, ParaSoft Corporation, Pasadena, CA, 1992.
- [18] C. L. Seitz, J. Seizovic, and W. K. Su, "The C Programmer's Abbreviated Guide to Multicomputer Programming", Technical Report Caltech-CS-TR-88-1, California Institute of Technology, Department of Computer Science, Pasadena, CA, Jan 1988.
- [19] J. Seizovic, "The Reactive Kernel", Technical Report Caltech-CS-TR-88-10, California Institute of Technology, Department of Computer Science, Pasadena, CA, Oct 1988.
- [20] A. Stettner and D. P. Greenberg, "Computer graphics visualization for acoustic simulation," *ACM Computer Graphics, Proc. SIGGRAPH '89*, Vol. 23, No. 3, July 1989, pp. 195-206.