Virtual Instruments in Virtual Rooms—
A Real-Time Binaural Room Simulation Environment
for Physical Models of Musical Instruments

Jyrina Huopaniemi1, Mari Karjalainen1,2, Vesa Välimäki1,2, and Torani Huumolinna1

1Helsinki University of Technology
Acoustics Laboratory
Otaniemi 5A, FIN-02150 Espoo, Finland
2CARTES
(Computer Arts Centre at Espoo)
Ahteenkujala 4, FIN-02100 Espoo, Finland
jyrina@huht.fi, mari@si.ka.hut.fi, vvs@si.ka.hut.fi, toma@huma.hut.fi

Abstract

In this paper, high-quality model-based sound synthesis of pitched and woodwind instruments is combined with room simulation and auralization techniques. The result is a real-time virtual threedimensional room where the listener and multiple virtual instruments can be moved. The system was designed using digital processors in a multiprocessing environment. Sound radiation patterns of musical instruments were measured and direction-dependent filtering was applied to instrument simulation models. Auralization is achieved by measuring and using head-related transfer functions. The binaural three-dimensional audio output of the virtual environment is directed to headphone listening. The environment is controlled by a mouse-opened user interface while the virtual instruments are played via MIDI.

1 Introduction

Digital waveguide and delay line modeling of wave propagation has been an interesting and rewarding field in computer music technology. Artificial reverberation generated by digital delay line networks has been studied for over three decades (see, e.g., [Breder, 1962]). Physical modeling of musical instruments using digital waveguides has shown to be one of the most important inventions in sound synthesis in the past two decades (Smith, 1987; Smith, 1992). Computer-based simulation of room acoustics has also adopted techniques based on digital delay lines. However, applications that combine different implementations of digital waveguides, such as sound-based sound synthesis attached to a virtual acoustical environment, are still quite rare.

This paper describes an implementation of a real-time virtual sound and acoustics simulator. As opposed to previously introduced virtual acoustical real-time environments (see e.g., [Peker et al., 1991]; [Asheim, 1993]; [Wenzel, 1994]), this paper combines several areas of interest and results in a new way of looking at virtual reality technology. This environment consists of moving sound sources that are physical models of musical instruments and a freely moving listener who is controlling the system. The models, a virtual acoustic guitar and a virtual flute, are played via MIDI and their movements are controlled from a keyboard. The radiation directivity of the instruments is taken into account by applying direction-dependent filtering. The listener-wearing headphones captures the binaural three-dimensional sound created by the environment. Auralization (Klitzner et al., 1993) is achieved by applying binaural digital filters designed from empirically measured head-related transfer functions (HRTF). [Braun, 1983] [Wightman and Kistler, 1989] (Moller, 1992).

The paper is organized as follows. In Section 2, the digital waveguide models for the acoustic guitar and the flute are reviewed. The major improvements over the former models is that now the sound radiation directivity of the instruments is incorporated.

In Section 3, the techniques for modeling room acoustics and binaural rendering are discussed. For room simulation we have adopted the image-source method [Allen and Berkley, 1977] combined with a method for approximating late reverberation. Measurements of HRTFs and implementations of the binaural digital filters are discussed with emphasis on real-time sound processing.

Section 4 presents the multiprocessing platform that is used for implementing the virtual environment. The QuickIC3000 DSP programming environment is discussed.

In Section 5, we reveal the secrets behind the designed virtual environment—how and where and
why the system is running, and what does it sound like.

The final section summarizes our results and makes suggestions for future work in the field of virtual sound reality.

2 The Physical Models

Sound synthesis based on physical modeling has recently become one of the most active fields of computer music research. The development of digital waveguide modeling techniques [Smith, 1992] has led to computationally efficient synthesis algorithms. The foremost goal in applying these methods to musical instrument modeling has been to achieve real-time sound synthesis on modern general purpose processors [Koistinen and Laine, 1991]. The research carried out at the Acoustics Laboratory of the Helsinki University of Technology has led to high-quality real-time synthesis algorithms of, e.g., the acoustic guitar [Koistinen et al., 1993] and the flute [Vilimaki et al., 1992].

2.1 The Guitar Model

The block diagram of the guitar model is depicted in Fig. 1. The output of the excitation filter $E(t)$ represents the signal produced by a plectrum or a plectrum touching a string. This signal is fed into the string model $S(z)$ that is presented in Fig. 2. This version of the string algorithm, derived from a waveguide model, is called the extended Karplus-Strong model. The feedforward loop (on the left in Fig. 2) models the effect of the pluck position. The length of the delay line $z^d$ determines the fundamental frequency of the synthetic tone and the loop filter $H(z)$ brings about the damping and dispersion effects.

The body model $B(z)$ is in principle a high-order filter that adds the most important resonances and anharmonicities of the body to the synthetic sound [Koistinen et al., 1991]. Due to the commutativity of the blocks depicted in Fig. 1 the impulse response $h(n)$ of the body may be incorporated in the excitation sequence as proposed in [Smith et al., 1991] and [Koistinen et al., 1993] [Smith, 1993].

The filter $R(z, \theta_i)$ is a single-input-multiple-output filter that models the radiation directivity of the guitar to different angles $\theta_i$ ($i = 0, 1, 2, ..., K$). Thus, the output of the guitar model is a vector of signals, that is $y(n) = [y_1(n), y_2(n), ..., y_K(n)]$. A practical implementation would include a small number of simple digital filters.

Fig. 1. The block diagram of the guitar model.

2.2 The Flute Model

The waveguide flute model is illustrated in Fig. 3. It can be divided into linear and nonlinear parts. The linear part represents the bore of the instrument and the reflection from the first open tone hole. This is similar to a resonator model of any woodwind instrument. The air jet model simulates the interaction of the excitation and the wave that propagates in the bore. This part of the system includes a nonlinearity which is characteristic of flute instruments or flute organ pipes.

The input signal $e(n)$ is a white noise sequence. The model produces two output signals, $y_1(n)$ and $y_2(n)$. The former corresponds to the noisy sound that radiates from the embouchure hole of the flute. The latter is radiated from the first open tone hole of the flute. For further details see [Vilimaki et al., 1992] or [Vuori and Vilimaki, 1993].

Fig. 2. The extended Karplus-Strong model.

2.3 Modeling of Sound Radiation Patterns of Musical Instruments

Most musical instruments exhibit complex sound radiation patterns. In string instruments different mode frequencies of the body have their own patterns such as monopoles, dipoles, or quadrupoles, and their combinations. In brass wind instruments the bell has increasing directivity towards high frequencies. In the flute there are inherently two points of radiation, the embouchure hole and the first open finger hole (or the end opening). In other wind instruments most of the radiation originates from a single hole, thus leading to a relatively omnidirectional pattern. Another noticeable factor of directivity is the masking by (and to a lesser degree the reflection from) the player of the instrument.

Computational modeling of the detailed directivity patterns is out of the capacity of real-time DSP sound synthesis. It is therefore important to find simplified models that are efficient from the signal processing point of view and as good as possible from the perceptual point of view. In the following we consider three strategies.
Fig. 4. Direction-dependent radiation of the acoustic guitar modeled with a 2nd order IIR filter. The amplitude scale is in db.

Directional Filtering

A direction-dependent digital filter may be attached to each path from the source to the listener. Moving and rotating sources can be modeled by changing the filter parameters of the paths in a proper way (e.g., the Leslie effect of a rotating loudspeaker can be simulated). The directional filtering method was studied for two instruments: the trumpet and the acoustic guitar. In the case of the trumpet the radiation response from the bell was measured to different angles by using an impulse-like excitation at the mouthpiece. We came to the conclusion that even first or second-order directivity filters give useful results thus leading to an efficient implementation.

Figure 4 depicts the modeling of direction-dependent radiation of the acoustic guitar (in the horizontal plane) relative to the main axis radiation. Shown in the figure are magnitude spectra for second-order IIR filters at azimuth angles 90°, 135°, and 180°. The reference magnitude spectrum at 0° is assumed to be H(2π1). The lowpass characteristic is noticeably increased as the relative angle increases. The measurement was carried out by exciting the bridge of the instrument by an impulse hammer and by registering the reference response at 0° and the related response in various directions. The measured reference and the directional response were fitted separately with first or second-order AR models. A simple division of the models was performed to get the pole-zero directivity filter.

The upper part in Fig. 5 depicts the impulse responses measured at azimuth angles 0° and 180°. The lower part shows the response from 0° filtered with a 1st order IIR directivity filter and the actual measured response at 180° azimuth. Note that the spectral slopes are nearly the same, as was expected. In this example the directional filter is

Fig. 5. Modeling of guitar sound radiation with a 1st order IIR filter. The relative amplitude scale is in db.

\[ H(z) = \frac{0.204z^{-2} - 0.143z^{-1}}{1 - 0.438z^{-1}} \]  

When zooming to the details of the lowest resonance modes we notice, as described in [Fletcher and Rossing, 1991], that the individual modes behave differently. To model such details a relatively high-order directional filter is needed. It is important to notice that due to the critical band frequency resolution an auditory smoothing may be applied to the responses before directional filter estimation. This helps to reduce the order of the filter.

Set of Elementary Sources

In this method the radiation pattern of an instrument is approximated by a small number of elementary sources such as monopoles or dipoles. We applied this approach to the flute where there are inherently two point sources of sound radiation, the embouchure hole and the first open tone hole. In general the method is computationally expensive if a large number of paths to the receiver is needed since each new elementary source adds a new set of path filters.
Direction-Dependent Excitation

In the case of commutative excitation (e.g., a plucked string instrument model) the directivity filtering may be included in the composite excitation in a way similar to the inclusion of the (early) room response [Smith, 1993]. The problem with this method is that as many instrument or string model instances are to be run in parallel as there are directions to be included. This limits the number of simulated directions, e.g., to the six main directions of the Cartesian coordinate system. Even then it is computationally very inefficient.

The considerations above as well as our experiments have shown that the directional filtering technique is normally the most practical one. A first or second-order filter approximation is often a satisfactory solution in a real-time implementation. In some cases such as the flute a set of elementary sources may be attractive.

3 Auralization and Modeling of Room Acoustics

The task of approximating a sound field is the basis of computational simulations of room acoustics. Several mathematical algorithms to accomplish this have been developed, most common being ray-tracing, image-source methods, and their hybrids. The goal in ray-tracing algorithms is to model the sound waves emitted from a sound source and follow their propagation and reflections from boundary elements; the walls, the ceiling and the floor. In the case of rectangular rooms it is, however, more convenient from the computational point of view to use the image-source method.

3.1 Image-Source Method

The image-source method is based on the well-studied theory of using phantom images of the desired source position in a bounded space. The Mth-order reflections of a sound source can be calculated by adding N images of the room and the source and calculating the length of the path between the two points. The sound radiating from the source and its images is attenuated due to three reasons: 1) absorption of the boundary materials, 2) the distance between the points of interest, and 3) the frequency-dependent air absorption. These factors have to be taken into account in accurate simulation of room acoustics. In DSP terms frequency-dependent filtering and attenuation can be accomplished with low-order digital filters. The image-source method that we have implemented assumes the modeled room to be rectangular and all the reflections to be specular.

Aspects in Real-Time Room Simulation

One remarkable drawback in the use of the image-source method is the computational explosion resulting from the calculation of higher-order reflections. For example, in a moderate-sized room (i.e., L x W x H = 5 x 3 x 3 m$^2$) an impulse response of 300 ms would need images of order $M = 20$ resulting in $N = n (a - 1)^{M-1} = 6.5^{20} = 1.1 \times 10^{14}$ separate image sources, where $n$ is the number of boundaries. When a limited time is allowed for the computation of the room impulse response, most of the described simulation methods become unavailable. As in our case, a realistic task even for a fast signal processor would be to calculate and update up to second or third-order reflections from a simple room using the image-source method. Thus, recursive methods for designing natural yet perceptually accurate reverberation have to be implemented. Luckily it has been shown [Baron, 1971] that about 80 ms of the first reflections is sufficient to create a spatial impression of reverberation. The late reverberation can be modeled artificially as is discussed in the following section.

3.2 Artificial Reverberation Using Recursive Methods

A framework for artificial reverberation was built by Schroeder using parallel comb filters and series allpass filters [Schoeder, 1962]. Most of the modern reverberation implementations exhibit complex combinations of both comb and allpass filter structures [Moore, 1979, 1983] [Gold and Chaine, 1991]. We have studied different aspects in reverberation modeling and ended up with a new structure, which is computationally efficient yet gains good results.

A novel implementation that clearly follows the guidelines from earlier reverberator designs has been implemented. The combined image-source and late reverberation filter is depicted in Fig. 6. The fundamental idea is 1) to use the direct sound and image-source method for accurate modeling of the first reflections, and 2) to use outputs from the direct and imaged sources as inputs to the late reverberation filter.

The input to the reverberator model is the direction-independent output from the musical instrument model. The upter delay line represents the time delay of the direct and imaged sounds. The direct sound and the imaged sounds are filtered with low-pass filters $g_0(n) - g_{N}(n)$ that include air absorption, boundary material absorption, and the radiation directivity of the musical instrument models.

The architecture used in the late reverberation model is similar to the guitar bridge filter model described in [Smith, 1993]. The idea is to feed each

Acoustics 458

ICMC Proceedings 1994
of the six recursive comb filter loops with the
negated sum from the five other loops. The blocks
$H_1(z) - H_6(z)$ represent a combination of pure
delay and a low-pass structure.

Our implementation of the reverberation filter
structure includes an allpass section $A(z)$, the output
of which is distributed as illustrated in Fig. 5.

3.3 Auralization

Auralization refers to binaural processing of
sound signals so that an illusion of three-dimen-
sional sound space is created [Klauer et al., 1993].
Real-time auralization involves not only digital sig-
nal processing but also good knowledge of the
mechanisms of human hearing and especially spatial
perception of sound. Executive studies of spatial
hearing (see, e.g., [Blauert, 1983]) have posed that
most of the spatial localization cues are caused by
three facts: 1) inter-aural time difference (ITD),
2) interaural amplitude difference (IAD), and 3) the
direction-dependent filtering by the outer ear (the
pinnae) and the torso. An approximation to the ITD
can be found in [Blauert, 1983] and is stated as:

$$\Delta t = \frac{D}{2} (\varphi + \sin \varphi)$$

where $\Delta t$ represents the difference in the path
length of the arriving sound, $D$ is distance between the ears
and $\varphi$ is the incident angle of the outgoing wave.

To avoid localization conflicts that often occur in
the median plane where ITDs and IADs are at their
minimum, accurate modeling of the direction-depen-
dent filtering has to be accomplished. One method

that has been successfully mastered in recent years is
based on measurements and implementations of head-related transfer functions (HRTF). This topic is
covered in the next section.

HRTF Measurements

A HRTF represents a free-field transfer function
from a fixed point to the entrance of the ear canal.
Measured HRTFs have distinctive characteristics
showing different torso and pinna-cavity reflections
and give information about human sound localiza-
tion. HRTF measurements were carried out to provide
data for the direction-dependent filtering in the aural-
ization model. The method that was used is described
in [Wightman and Kistler, 1989]. The probe micro-
phone transfer function can be described in the fre-
cy domain as

$$Y(f) = X(f)M(f)$$

where $Y(f)$ represents the probe microphone's
response of the test signal driven to a loudspeaker
$X(f)$. $M(f)$ represents the free-field-to-ear-canal
transfer function, $L(f)$ the loudspeaker transfer func-
tion, and $M(f)$ the microphone transfer function.
A similar measurement is made using headphones
resulting in a transfer function

$$Y(f) = X(f)H(f)M(f),$$

where $H(f)$ represents the headphone-to-ear-canal
transfer function. When both probe transfer functions are set to equal, the desired filter transfer function is given by

$$T(f) = \frac{L(f)X(f)}{H(f)}$$

It is clearly seen that the frequency characteristics of
both the loudspeaker and the headphones are uncompens-
tated in this form. From this equation the loud-
speaker transfer function remains to be compensated
to get the actual HRTF for headphone reproduction.
Cubalizations are performed by deconvolution in
the frequency domain.

Measurement Equipment

The equipment used for measurements were espe-
cially designed and built for this purpose. The probe
microphone system consisted of a Sennheiser KE4-
211-I condenser microphone coupled to a polyethylene
probe tube with an outer diameter of 1.77 mm. A set of two microphones with one stereo preamplifier were built to allow measurements from both ears simultaneously.

The measurement software was programmed in
Common Lisp using the QuickSig [Karpilainen, 1996] software. The test signal used was a flat speci-
HRTF Filter Design

A reconstruction of FIR filters from measured HRTFs was performed. A more detailed analysis of the HRTFs will be carried out in near future. An example set of 128-tap FIR filters were calculated. Figure 7 shows the magnitude spectra of the filters from azimuth positions 0°, -45°, and -135°.

4 Multiprocessor DSP Environment

A large amount of computational power is required to achieve real-time performance for binaural room simulation and model-based sound synthesis. The QuickC30/C40 is a multiprocessor DSP environment developed at the Acoustics Laboratory of the Helsinki University of Technology (Karjalainen, 1992). The system contains several signal processors and allows for complicated real-time applications to be realized.

4.1 Hardware

The QuickC30/C40 multiprocessor system hardware consists of multiple signal processor cards and an Apple Macintosh host computer. The host computer communicates with the processors through a parallel digital I/O interface and a handshaking logic. The multiprocessor system is based on Loughborough Sound Images’ MDC40S modules which adhere to Texas Instruments TIM-40 specification (Weir, 1992). The latter specification is in turn based on the Texas Instruments TMS320C40 floating-point signal processor [Texas Instruments, 1991], that has been developed for parallel processing. The processors communicate with each other through communication ports. We have designed special mother cards to simplify the linking of the MDC40S modules. The environment includes 16 or 18-bit AD/DA converters by Crystal Semiconductor.

4.2 Software

The QuickC30/C40 environment has been implemented entirely within the Common Lisp Object System (CLOS) for the Apple Macintosh (Karjalainen, 1992). The software of the multiprocessor system consists of a message passing protocol based on interrupt routines supporting parallel processing for the communication between the signal processors and the host computer. The system has two separate buses for communication between the C40 processors, one for short messages and another for block transfers as illustrated in Fig. 8.

5 The Virtual Environment

The designed environment was built and programmed to function on a Apple Macintosh Quadra 950 computer with two TMS320C40 signal processors. The room simulation and auralization software as well as the musical instrument models were programmed in Common Lisp using its object-oriented extensions CLOS and the QuickC30/C40. The current sampling rate used is 22.05 kHz.

Fig. 7. Measured HRTFs from three azimuth angles. The relative amplitude scale is in dB.

Fig. 8. The QuickC30/C40 multiprocessor system consisting of two TMS320C40 signal processors.

Fig. 9. The virtual environment object hierarchy.
5.1 The Building Blocks

The virtual room model consists of three separate building blocks, objects, that can be instantiated separately or together forming the complete system (Fig. 5). The room object contains the basic definitions for the dimensions and for the positions of the listener and the source objects. It also creates the image sources of the designated sound source for accurate modeling of first reflections. The sources slot of the object refers to the musical instrument model (guitar or flute), which will be instantiated to run on one of the C40s.

The next class in the hierarchy, room-with-window, inherits the properties from the room object and adds routines for manipulating the graphical source and listener objects, and for adjusting the room size.

The two previously described object classes are defined and accessed in the host computer’s memory. They provide links to the low-level signal processor environment, and the links are established in the following object class.

The auralized-room-with-window class inherits the room and graphics properties from previous classes. The communication routines between the two signal processor and the host computer are controlled within this object class. An example of creating a room simulation object is shown in Fig. 10.

Event Controlling

The graphical events are the controllers of the calculations performed in the signal processors. Moving or reorienting the graphical objects on the screen interrupts the processors by updating the source, source image and listener positions. The needed information for auralized sound processing is: 1) the distance between the source or its image and the listener, 2) the relative elevation angles, and 3) the relative azimuth of the listener and the sound source. The MIDI interface is programmed according to [Karlajainen et al., 1993].

6 Summary and Future Work

A virtual binaural room simulator for physical models of musical instrument was presented. The system produces virtual three-dimensional audio output for stereo headphone listening in real-time. Radiation directivity of the musical instruments has been incorporated by applying direction-dependent low-order digital filters. Room simulation is performed with a combined image-source and late reverberation model. Binaural processing is carried out using digital filter approximations of measured HRTFs.

Future areas of interest in this research project include accurate non-real-time modeling of the environment and a 3-D graphical network interface to the room simulation system. We are also planning further investigations on the modeling of musical instrument radiation patterns.

Acknowledgments

The authors would like to acknowledge the persons taking part in numerous HRTF experiments and measurements. This work has been financed by the Academy of Finland.

References


