

# Real-time acoustics modeling in games

Joachim Bengtsson  
*Blekinge Institute of Technology*

Document revision 2. Last edited at 15 aug 2007.

*Abstract* This paper evaluates algorithms for modeling the acoustic properties of sounds in real time in games, and in particular evaluates J. Borish' "Extension of the image model to arbitrary polyhedra" [1] through implementation inside the RMS game engine. REST IS TBD.

# Table of Contents

<b>Introduction</b>	<b>5</b>
Hypothesis	5
Methodology	5
<b>Methods of Acoustic Modeling</b>	<b>6</b>
Method: Wave-based	6
<i>Viability for real-time applications</i>	6
Method: Ray-tracing	6
<i>Viability for real-time applications</i>	6
Method: Image model	6
<i>Viability for real-time applications</i>	6
Methods used in games today	6
<i>Thief</i>	6
<i>Half-Life 2</i>	6
<i>Company of Heroes</i>	6
<i>???</i>	6
<b>Studying the Image Model and its Implementation in RMS</b>	<b>7</b>
Architecture	7
<i>Generation</i>	7
<i>Acoustics Modeling</i>	7
<i>Auralization</i>	7
Implementation	7
Subjective Opinions about the Quality of the Output	7
Performance Measurements	7
<i>Latency</i>	7
<i>Processing time</i>	7
<b>Conclusion</b>	<b>8</b>
Analysis	8
<i>What do my numbers say?</i>	8
<i>Bad performance inherent to the IM or my implementation?</i>	8
Possible enhancements	8
<i>Sound generation</i>	8
<i>Late reverberation</i>	8
<i>Binaural reproduction</i>	8
<i>Adherence to room surface materials (material reflection filters)</i>	8
<i>Transfer through solids</i>	8
Wrap-up	8
<b>References</b>	<b>9</b>



## Introduction

It's common for today's games to use the geometry surrounding the listener (that is, the player) to calculate occlusion and some dampening, but they still often use more traditional methods for applying echo and reverb, through DSP effects instead of real physical simulation; mapping regions in space to presets of DSP settings.

As gaming hardware and commonplace PCs grow ever faster, however, and get multiple execution units (such as in the Intel Core Duo/Quad), it is becoming more and more realistic to do a more sophisticated modeling of the acoustics of the player's environment, simulating the environment in real-time without "cheating" with DSP.

Moreover, creating a "real", realistically-simulated sound environment has been a dream for me for many years; I first envisioned it maybe seven years ago, although I could hardly code then.

**This essay** will go through primarily the Image-Source Model of acoustics modeling, but also talk about some other possible modeling methods and compare them to the Image-Source Model. It'll see what other games use. The essay will go on to discuss the architecture and implementation of a game sound engine that uses the Image-Source Model in real-time, and its subjective quality and objective performance. Finally, it will conclude whether this is good enough, and list some possible enhancements to the engine.

I will try to answer the following questions:

- ◆ Is real-time sound ray tracing even remotely possible?
- ◆ What methods of acoustic modeling are used in games today?
- ◆ What methods are available (apart from [1] and ray tracing) for DSP-free acoustic modeling?
- ◆ How can the image model be implemented inside the RMS game engine<sup>1</sup>?
- ◆ Will the image model perform within realistic performance requirements (that is, be able to run concurrently with the rest of the game engine while still having a smooth, fluid gaming experience)?

I expect to find that real ray-tracing of the environment is still not feasible even with current hardware. Furthermore, I expect the algorithm in [1] to suit the task performance-wise and quality-wise to work in the sound engine in RMS.

## Hypothesis

With proper limits, the Image-Source Model can model the acoustic environment in real-time in a first person shooter game.

## Methodology

---

<sup>1</sup> See glossary for "RMS Engine"

## Methods of Acoustic Modeling

### Method: Wave-based

Description of wave-based models.

*Viability for real-time applications*

Not viable. Too slow. However, see 'viability' on Ray-tracing.

### Method: Ray-tracing

Description of how to ray trace sounds.

*Viability for real-time applications*

Not viable in real-time, but might be integrated with the pre-compilation of levels in FPS games.

### Method: Image model

Description of the image model.

*Viability for real-time applications*

Actually gives a *complete* model, which really isn't viable in real time. Giving it limits will make it viable, but then we'll lose out on the late reverberation.

## Methods used in games today

*Thief*

*Half-Life 2*

*Company of Heroes*

???

# **Studying the Image Model and its Implementation in RMS**

## **Architecture**

*Generation*

*Acoustics Modeling*

*Auralization*

## **Implementation**

### **Subjective Opinions about the Quality of the Output**

### **Performance Measurements**

*Latency*

*Processing time*

## **Conclusion**

### **Analysis**

*What do my numbers say?*

*Bad performance inherent to the IM or my implementation?*

### **Possible enhancements**

*Sound generation*

*Late reverberation*

Through ray tracing parameters in a pre-compilation stage.

*Binaural reproduction*

*Adherence to room surface materials (material reflection filters)*

*Transfer through solids*

### **Wrap-up**

## References

- [1] J. Borish, Extension of the image model to arbitrary polyhedra, Acoustical Society of America Journal Vol. 75, 1984.
- [2] L. Savioja, Modeling Techniques for Virtual Acoustics, <http://www.tml.tkk.fi/~las/publications/thesis/>, 1999

## Appendix A: Glossary

**RMS Engine** The RMS engine is the game engine used in Friendly Stapler's "Rymdvar-  
elser Mot Soldater" (third year group project in BTH's Game Program-  
ming's class of 2004, in which I work). See also [friendlystapler.se](http://friendlystapler.se).