Sound Localization and Multi-Modal Steering for Autonomous Virtual Agents

Yu Wang[*]   Mubbasir Kapadia[†]   Pengfei Huang[*]   Ladislav Kavan[†]   Norman I. Badler[*]
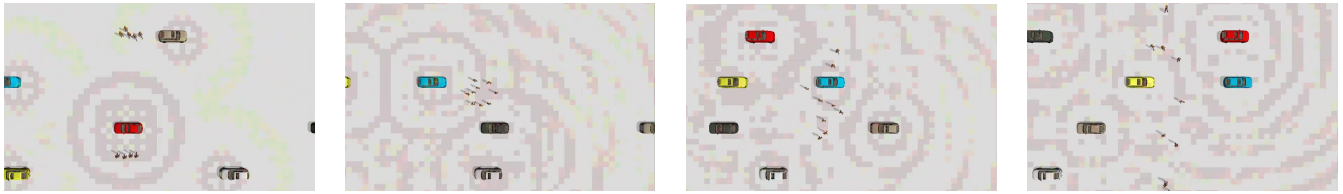
University of Pennsylvania

Figure 1: *Blind agents relying solely on sound localization and sound-driven collision avoidance while navigating along a highway with crossing vehicles that emit sounds.*

Abstract

With the increasing realism of interactive applications, there is a growing need for harnessing additional sensory modalities such as hearing. While the synthesis and propagation of sounds in virtual environments has been explored, there has been little work that addresses sound localization and its integration into behaviors for autonomous virtual agents. This paper develops a framework that enables autonomous virtual agents to localize sounds in dynamic virtual environments, subject to distortion effects due to attenuation, reflection and diffraction from obstacles, as well as interference between multiple audio signals. We additionally integrate hearing into standard predictive collision avoidance techniques and couple it with vision to allow agents to react to what they see and hear, while navigating in virtual environments.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation;

Keywords: Virtual agents, artificial life, acoustics, localization, steering

Links: ◆DL 🅿PDF

1 Introduction

As the visual and simulation fidelities of interactive applications continue to reach new heights, there has been a growing interest to fill the void in an equally important sensory modality – *hearing*. This has led to many exciting recent contributions for synthesizing [O'Brien et al. 2002; James et al. 2006] and propagating [Raghuvanshi et al. 2009] sounds in complex 3D virtual en-

[*]{wangyu9, pengfei, badler}@seas.upenn.edu

[†]{mubbasir.kapadia, ladislav.kavan}@gmail.com

vironments, enabling users to perceive high-quality audio content. However, autonomous agents that populate these environments and interact with human-controlled avatars lack an appropriate mechanism to perceive and react to acoustic signals, which limits the perceived realism of their behavior and breaks immersion.

Identifying where sounds originate (sound localization), understanding how it impacts an agent's movement (sound-driven navigation and collision avoidance), and fusing it with visual perception can greatly enhance the behavioral repertoire of NPC behavior. For example, an agent can hear the footsteps of a player following from behind, localize enemy gunfire, or use hearing to predict the spatial location of other entities in the dark, which can significantly impact its response.

There has been a recent surge in contributions for the synthesis [Bonneel et al. 2008] and propagation [Raghuvanshi et al. 2010] of sound signals in virtual environments. However, traditional approaches [Monzani and Thalmann 2000] rely on distance-based heuristics to impact the behavior of autonomous agents in response to auditory signals. This simplified hearing model produces artifacts because the influence of obstacles on sound propagation is not considered: for instance, two agents separated by a wall should not hear each other, even though they are close to each other.

The motivation for this work is to combine a physically accurate model for sound propagation and localization, and integrate it into agent navigation and collision avoidance. Sound signals are accurately propagated in the environment while accounting for degradation to due to absorption, reflection, diffraction, and mixing of sound signals. The pressure and gradient field of the propagated sounds are computed to find its local directional flow, and the integration of several detectors per receiver is used to localize the sound signal. A smooth and continuous tracking of sound sources is obtained by applying a Kalman Filter [Thrun et al. 2005] to the predicted sound positions.

Using the predicted position and velocity of different sound signals, we introduce *sound obstacles* which are generalized velocity obstacles [Wilkie et al. 2009] for objects in the environment which an agent hears, but cannot see. We integrate sound obstacles into a traditional vision-based steering approach [Shao and Terzopoulos 2005; Yu and Terzopoulos 2007] to simulate autonomous agents that integrate hearing into navigation and goal-directed collision avoidance. If no visual information is used, we can simulate the behavior of a virtual blind agent. When combined with visual perception, we demonstrate autonomous agents that exploit hearing for objects that are currently not in their line of sight, to greatly enhance their behavior in dynamic environments. We demonstrate the benefit of

sound localization by integrating it into the steering response of an agent, but it can be potentially used to impact decision-making at all levels of cognition. The main contributions of this paper are as follows:

- We introduce the ability of autonomous virtual humans to predict the position and velocity of sound-emitting objects based on what it hears, subject to sound propagation and distortion in dynamic virtual environments.

- We present a multi-modal steering platform that integrates hearing into a traditional vision-based model, allowing agents to predict and react to the cumulative presence of objects that they may hear or see.

2   Related Work

Computational Acoustics. The theory of wave propagation is well established in classical physics. Sound is governed by the wave equation, which is a second-order partial differential equation. Techniques for sound simulation can be roughly classified into numerical acoustics (NA) and geometric acoustics (GA).

Numerical acoustics is directly solving the wave equation. Classic methods include finite difference time domain (FDTD), finite element, and boundary element methods. FDTD uses time domain difference to approximate derivative, and it can handle a wide frequency range with a single simulation run. The finite element method uses an irregular discretization, allowing it to adapt to complex boundaries [Ihlenburg 1998]. Boundary element method only requires a mesh of the boundary of the domain [Ciskowski and Brebbia 1991]. Numerical methods are accurate but very costly: for example, for FDTD every wavelength should have 6-10 samples to give an accurate result [Mehra et al. 2012], which makes it very expensive.

The Transmission Line Matrix method (TLM) is very popular in electromagnetic wave propagation, and it is also applied to the simulation of sound waves [Kristiansen and Viggen 2010; Kagawa et al. 1998]. TLM can be regarded as a simplified numerical method. However, different from FDTD, the ratio of λ (wavelength of the sound we are simulating) to h (grid spatial step) is a constant determined by the model, i.e., for a given grid resolution we can only simulate sound with certain frequency. Although the TLM method features some limitations, we argue it is an adequate approximation in the context of virtual agents. TLM is simple, easy to implement and parallelize.

Geometric acoustics is a high frequency approximate method for sound simulation. If the wavelength is much smaller than object dimension (such as light), the wave equation can be approximated by raytracing, which assumes that sound propagates as rays of energy quanta. Typical geometric acoustics includes volumetric tracing (beam tracing and frustum tracing) [Funkhouser et al. 2004; Antonacci et al. 2004] and image source method [Allen and Berkley 1979]. Compared with numerical acoustics, it is computationally efficient and there are lots of methods to accelerate ray tracing in graphics. However, geometric acoustics cannot fully simulate low-frequency phenomena such as diffraction [Kristiansen and Viggen 2010], and methods such as edge-diffraction [Funkhouser et al. 1998] have been proposed to approximately capture these lower order effects. Precomputed Acoustic Transfer (PAT) has been used to accelerate both sound synthesis [James et al. 2006] and propagation [Raghuvanshi et al. 2010].

Sound Localization. Sound source localization (SSL) draws a lot of attention from biomedical scientists, physiologists, engineers and computer scientists. Distance estimate can be achieved by mea-
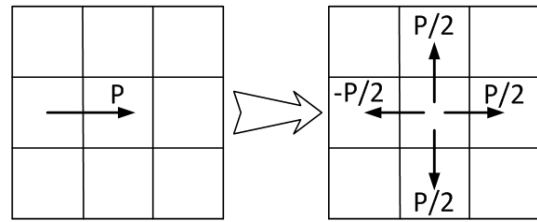
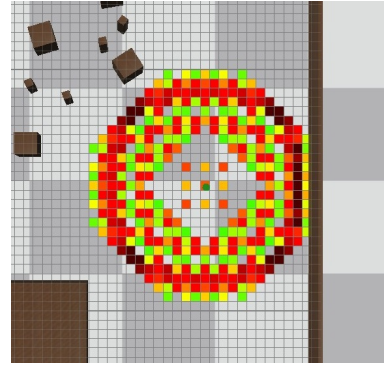

Figure 2: *An illustration of TLM method.*

Figure 3: *A snapshot of 2D-TLM sound propagation model. P is a sound packet in one grid.*

suring sound intensity and spectrum, and during the process a prior knowledge about the source's characteristics of radiation is needed [Strumillo 2011]. The mechanism of human's ability to determine the location of nearby sound sources is not fully understood [Martin 1995]. Human depends on a number of anatomical properties of the human auditory system, including *interaural intensity difference (IID)*, *interaural time difference (ITD)*, and *directional sound filtering* of the human body. An artificial robust localization system demands different approaches [Strumillo 2011], and often uses pressure sensors arrays. In the area of robotics, one of the most widely used method for the passive localization of acoustic source is based on the measurement of the *time delay of arrival (TDOA)* of the source signal to receptor pairs [Huang et al. 1997; Strumillo 2011]. By locating three sensors and recording the time difference of sound arriving, it is easy to calculate the position of sound source analytically. Instead, our algorithm makes use of local sound packets information that the virtual human perceive to determine the direction and distance of the sound source, and it also provides clues for the confidence of localization.

Robot Localization. While the robot localization problem often refers to the (active) self-localization of robots, different from our (passive) source localization problem, there are many shared ideas. Particle Filter and Kalman Filter are cornerstones of many such algorithms [Thrun et al. 2005]. Extended Kalman filter is combined with landmarks to tackle the simultaneous localization and mapping (SLAM) problem [Dissanayake et al. 2001].

Vision-based Steering. There is a a vast amount of literature in goal-directed collision avoidance for autonomous agents and we refer the readers to extensive surveys [Pelechano et al. 2008; Thalmann and Musse 2013; Kapadia and Badler 2013]. Steering techniques use reactive behaviors or social force models [Helbing and Molnar 1995; Pelechano et al. 2007] to perform goal-directed collision avoidance in dynamic environments. Predictive approaches [Paris et al. 2007; Van den Berg et al. 2008; Singh et al. 2011a] and local perception fields [Kapadia et al. 2009] enable an agent to avoid others by anticipating their movements. Recent work applies accelerated planning techniques [Singh et al. 2011b; Kapadia
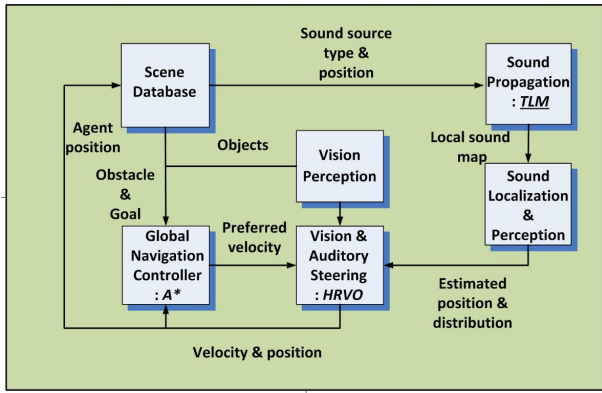
Figure 4: *Framework of our agent perception and steering.*

et al. 2013] to solve challenging deadlock situations in crowd interactions.

Reciprocal Velocity Obstacle (RVO) [Van den Berg et al. 2008] is a popular method both for robot navigation and agent simulation. By introducing the concepts of reciprocal velocity obstacle, the method calculates geometrically collision-free velocity set for the agent and pick the best velocity (closest to preferred velocity) in the set. *Hybrid Reciprocal Velocity Obstacle* (HRVO) [Snape et al. 2011] is an extension of original RVO method, and accommodates noise in visual information which is useful for robots. The work in [Ondřej et al. 2010] proposes a synthetic vision-based approach to collision avoidance. The work in [Shao and Terzopoulos 2005] integrates a vision model to drive reactive collision avoidance, navigation, and behavior for autonomous pedestrians.

3 Framework Overview

Figure 4 illustrates an overview of the framework. Sound signals are propagated in a dynamic virtual environment to capture various acoustic effects including attenuation, reflection, and diffraction. Agents equipped with hearing perceive the sound pressure and gradient at their locations, which is used to compute the predicted position and velocity of the sound emitting objects (Section 4). Finally, a multi-modal steering framework integrates visual and auditor information to enable autonomous agents to predict and react to the presence of dynamic entities in the virtual environment that they may hear or see (Figure 11).

3.1 Sound propagation model

A computational method for simulating sound must satisfy limits on computation time and memory [Mehra et al. 2012], while accounting for relevant acoustic properties such as attenuation and diffraction of sound signals in order to make them feasible for interactive applications. We adopt a planar model that uses the Transmission Line Matrix Method (TLM) [Kagawa et al. 1998; Huang et al. 2013] for sound propagation in complex dynamic environments. Even though propagation is planar, the sound can be propagated across different planes at different heights, to produce the effect in a 3D environment, and our proposed approach can be extended to 3D intuitively. We briefly describe the TLM method below and refer the readers to a comprehensive overview for more details [Kristiansen and Viggen 2010].

Sound is governed by the wave equation or, equivalently, Huygens principle, which states that "every point of a wave frontier can be considered as a source of secondary wavelets known as a sub-source which spread out in all directions". The TLM model consists of a
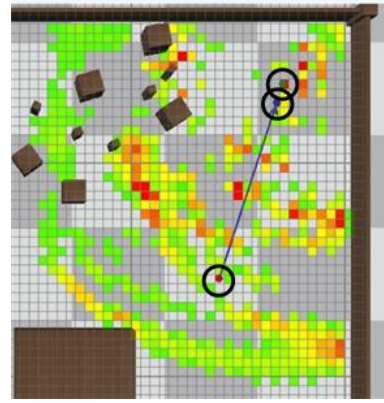

Figure 5: *An illustration of sound localization, the green mark is sound source, the red one is the agent (receiver), and the blue one is the estimated position of sound source(output of our algorithm). Markers have been circled.*

mesh of interconnected nodes. All cells are updated in parallel, and the update of a cell is determined only by cells in its vicinity [Kristiansen and Viggen 2010]. The update rule is shown in Figure 2. Based on Huygens principle, the energy of a directional incident pulse with an amplitude scatters to four directions.

One grid in TLM contains several packets, and each packet has one of four possible directions$\{N, S, W, E\}$. Initially, the packets emanate from a sound source in all four directions. At each iteration, the packets are updated according to rule shown in Figure 2. The sound packets around the receiver will be subsequently used as input to sound localization [Huang et al. 2013]. The output of TLM method that will be used in next section is what we called a "sound map". The sound map is analogous to an image but with four channels per pixel, corresponding to the directions.

4 Sound localization

Psychology experiments [Loomis et al. 1998] have investigated auditory perception and showed that the mean error for different target azimuths is usually less than $5°$ for audition perception, and a 7-meter change in target distance (from 3 to 10 m) could produce a change in mean indicated distance of 5.4 m for vision and of only 3.0 m for audition. The experiment demonstrates that the perceived egocentric distance of auditory perception exhibits more error than that of visual perception, and this difference between the two sensory modalities needs to be captured to simulate believable autonomous virtual humans or agents. The input from the TLM-based sound propagation is used to localize sound emitting objects using a binaural localization model, which is used to supplement visual information to produce a more complete mental model of an agent's surroundings. Before localization we assume the agent has already distinguished sounds from different sources.

4.1 Possible Localization Clues

We firstly examine some possible clues for localization that are used by human or robot.

Time Delay of Arrival. The TDOA (time delay of arrival) method measures the time delay (distance) differences between several differently located detectors to the same source to localize sound source and is greatly impacted by the accuracy of the time delay measurement. TDOA might be suitable for ray tracing based sound model where the sound paths are explicitly calculated; however it
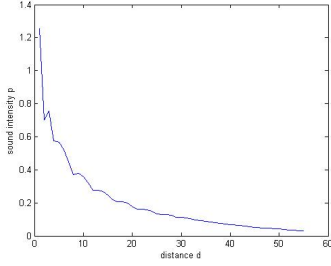
Figure 6: *Measured sound intensity as a function of distance in TLM model.*

cannot be integrated with our sound propagation model (TLM) because if the representation of sound grid is rough, the result of T-DOA will be very inaccurate. For example, if the distance between two receptors pairs is 4, due to triangle inequality there is only 9 possible time delays (from -4 to 4).

Binaural Hearing. The sound intensity p perceived by the agent is computed as follows:

$$p = \frac{log(I)}{d^2} \tag{1}$$

Our experiment validate the relation between the sound intensity p and distance from sound source d in TLM model shown in Figure 6. Thus, like real human, the nuance of intensity between two ears also provides clues for virtual agent's localization. However, like time delay, such feature lacks accuracy in TLM model; for instance we can see from Figure 6 that in TLM model the function is not exactly monotone.

Field Gradient of First-arriving Sound. This is what we use for localization. In psychology, it has been proposed that humans prefer the direction of the first-arriving sound or so-called direct sound for localization, which arrives at a given position before any reverberation effects [Litovsky et al. 1999; Martin 1995]. For algorithm design, the advantage of using only first-arriving sound is that echo filtering and signal processing is not needed. The cumulative vector $\mathbf{m}$ that we will see later actually corresponds to the sound pressure field of first-arriving sound; it is the intuition behind the detector that we will introduce.

4.2 Sound Flow Detection

We detect the direction of the sound flow by tracing the local flow of the sound wave energy, to compute the sound field gradient which reveals the position of the source. First, we define the center of the sound energy as the weighted average position of sound wave energy using the energy values as the weight, similar to calculating the center of an object using their mass or gravity as weight. In other words, if we consider the sound packets to be virtual balls with mass (energy), we can calculate the "momentum" of the region as follows:

$$\mathbf{m}_t = \sum_{\forall p_i \in \mathbf{P}} \mathbf{v}_i \cdot E_i \tag{2}$$

where $\mathbf{P}$ are the set of sound packets in the region, $\mathbf{v}_i$ and E_i are velocity and energy of the sound packet p_i, and $\mathbf{m}_t$ represents the momentum vector in the region at time t.

Experiments show $\mathbf{m}$ can reflect the opposite direction of sound source: although every sound packet's velocity only has four possible directions $\{N, S, W, E\}$, our experiments show that sound packets in one gird is enough to give an acceptable result and produce no noticeable error.

If we use $\mathbf{m}_t$ to denote the vector that we obtained at time t, we can make it smoother and more robust by using cumulative vector:

$$\mathbf{m} = \sum_{t=T}^{T+t_w} \mathbf{m}_t \tag{3}$$

where $\mathbf{m}$ is the cumulative vector, and T is the time when the first sound packet is perceived by the agent. The length of sampling period of perceived signal is t_w (time window). When t_w is short, we only collect the sound packets propagating along the shortest path from sound source to the agent, thus producing an echo-free effect without any reverberation. In TLM model, we find for $t_w = 4 \sim 10$ generally gives good results. $t_w \leq 3$ is too sensitive and sometimes does not give the correct answer.

The direction of the sound source is computed as:

$$\theta = atan2(-m_y, -m_x) \tag{4}$$

where m_x, m_y are x, y component of vector $\mathbf{m}$. We have two minuses here because sound momentum is always in the opposite direction of source.

4.3 Ensemble of Detectors

Let there be n sound detectors, and each of them can localize the source on a line. Our task is to integrate the outputs of all detectors. We build a very simple probabilistic model to tackle this problem. Assume that detector D_i localizes the source on the line l_i intersecting at point (x_i, y_i) with a slope of $\tan \theta_i$, where (x_i, y_i) is the position of D_i, and θ_i is as defined in Eq. 4:

$$\sin \theta_i(x - x_i) - \cos \theta_i(y - y_i) = 0$$

The distance of an arbitrary point (x,y) to the line l_i is given by:

$$d_i(x, y) = \| \sin \theta_i(x - x_i) - \cos \theta_i(y - y_i) \|$$

If we have one observation taken from D_i, we assume the probability distribution of the source is given by a Gaussian distribution, which works seamlessly with Kalman Filter. If we have multiple detectors D_i (i from 1 to n), we simply multiply the probabilities together.

$$P(x, y \mid D_i) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{d_i^2(x,y)}{2\sigma^2}}$$

$$P(x, y \mid D) = \prod_{i=1}^{n} P(x, y \mid D_i) = \frac{1}{(\sqrt{2\pi}\sigma)^n} e^{-\frac{1}{2\sigma^2} \sum_{i=1}^{n} d_i^2(x,y)}$$

The output estimate of source position (x_o, y_o) is given by maximizing the probability:

$$(x_o, y_o) = \arg \max_{(x,y)} P(x, y \mid D) = \arg \min_{(x,y)} \sum_{i=1}^{n} d_i^2(x, y)$$

$$\begin{pmatrix} x_o \\ y_o \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{n} \sin^2 \theta_i & -\sum_{i=1}^{n} \sin \theta_i \cos \theta_i \\ \sum_{i=1}^{n} \sin \theta_i \cos \theta_i & -\sum_{i=1}^{n} \cos^2 \theta_i \end{pmatrix}^{-1} \begin{pmatrix} \sum_{i=1}^{n} (\sin^2 \theta_i x_i - \sin \theta_i \cos \theta_i y_i) \\ \sum_{i=1}^{n} (\sin \theta_i \cos \theta_i x_i - \cos^2 \theta_i y_i) \end{pmatrix}$$

We only use sound flow detector here; we could also include time delay clues but our experiment shows that it does not contribute to the accuracy of localization in TLM sound model, due to its lack of accuracy according to the previous analysis. Employing more detectors can also increase the robustness of the algorithm but also increase the computational cost; in practice, we choose $n = 3$.

At a first glance it might be strange to assume that one agent has three or even more "ears". However, if we only use two detectors ($n = 2$), localization fails when the two detectors and the source are collinear, which happens quite often. While integrating more detectors (array of sound detectors) generally works better, our experiment shows three detectors demonstrate satisfactory robustness and accuracy.

4.4 Confidence of Sound Localization

Let us emphasize that it is not always possible to localize the sound source. For example, if the sound is generated by 100 tiny sound sources distributed in different places in the space, there does not exist a single sound source position. To reflect this, we introduce a measure of the confidence of the sound localization.

In certain conditions the localization is ambiguous and the localized position itself is insufficient to describe human's localization, so it is necessary to introduce the *confidence* of sound localization in auditory perception. When the auditory information is very fuzzy, which might be caused e.g. by multiple reverberations, the localization is less believable, and thus the weight or priority of this sound source should be small.
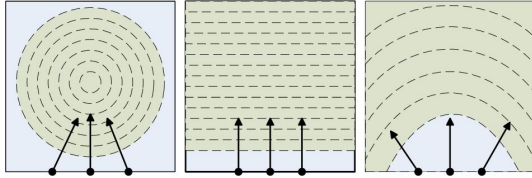

Figure 7: *Several possible wavefronts. Dotted lines indicate the shape of wavefront.*

Figure 7 shows several cases of a wavefront. The left image shows the situation where the sound source can be approximately regarded as a single source. In this case, sound localization is well-defined. The middle image shows the situation where the equivalent sound source is infinitely far. The right image shows the situation where the wavefront tends to "converge", which is impossible for a single source where no obstacle exists. However, it is possible when there are obstacles. Only in the left image the output of the localization is "legal", i.e., the localization is well-defined; in the right image, the detectors will converge to the opposite direction.

The situation when there are obstacles in the map is equivalent to the situation that sound is produced by a lot of tiny sources (subsources) distributed at different places. If such sub-sources are located in almost the same direction, our algorithm could still give an estimate of an equivalent single source. If such sub-sources are more widely distributed, it is only possible to give an estimate of the direction of the equivalent single source: we do this by simply averaging all detectors' direction estimate together. The worst case is that sound comes equally from all directions around the agent; in this case it is impossible to localize the sound source.

Figure 8 illustrates the relative confidence of sound localization due to the presence of different obstacle configurations. The momentum vector $\mathbf{m}$ estimates the direction of a sound source, and its magnitude could also be useful. When the sound sources or sub-sources share similar directionality, the magnitude of $\mathbf{m}$ will be strengthened; when they are in different directions, it will be weakened. For a single source, according to Huygens' principle, a greater confidence value means that the directions of all of the sub-sources are similar, or obstacles have little influence.

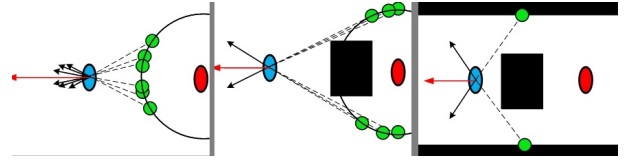Sound directionality is hard to measure for TLM model, because

Figure 8: *An illustration of localization confidence. The red mark and the blue mark are sound source and receiver respectively, and the green marks are major sub-sources and each of them has a contribution (black vector) to the total "momentum" vector (red). From left to right, we can see the sub-sources become more disperse and confidence decrease from 1 to 0.7, and to 0.5.*

sound paths are not explicitly present in the simulation. Directionality or localization confidence is much easier to measure in models based on ray tracing, where sound paths are maintained, so we know the confidence by comparing whether these paths are similar in direction. In practice, for TLM model we could use the ratio of magnitude of the aggregate momentum vectors to the sound intensity calculated by Eq.1 as the confidence metrics, but it assumes the agent already knows sound source intensity and real distance. Although there seems to be no reasonable solution to judge localization confidence, our discussion leads to the following algorithm:

1. **Agent successfully locates sound source.** If all detectors converge to a legal position, output both the distance and direction.

2. **Agent only infers direction of sound.** If all detectors output similar directions while converging to an illegal position (Figure 7 right case), output only direction.

3. **Unsuccessful localization of sound.** If detectors output contradictory directions, there is no output.

4.5 Tracking of Sound Sources

Sound localization algorithm will give output periodically, which must be translated to a continuous estimate of the source positions, given our past and current observations. There are several methods we might use for tracking; Particle Filter and Kalman Filter are the most widely used. Particle Filter uses a group of particles (e.g. 200 particles) to represent the spatial belief distribution of the sound sources. However, the computation cost of particle filter is high especially when we have multiple sources and multiple receiver of sound. Kalman Filter is an alternative choice for localization which assumes that the state of an object updates linearly and we obtain an observation every time step.

$$X_t = A_t X_{t-1} + B_t u_t + \epsilon_t$$
$$Z_t = C_t X_t + \delta_t$$
$$X_t \sim \mathcal{N}(\boldsymbol{\mu_t}, \boldsymbol{\Sigma_t}), \epsilon_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{R_t}), \delta_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{Q_t})$$

where X_t is the state space; Z_t is the observation for sound source we get from the previous sections, and ϵ_t and δ_t are noises of state transfer and observation. For example, if we assume that the source moves linearly, we can build the following motion model:

$$X_t = \begin{pmatrix} x \\ y \\ v_x \\ v_y \end{pmatrix} \qquad A_t = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$Z_t = \begin{pmatrix} Z_x \\ Z_y \end{pmatrix} \qquad C_t = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

We have no external control u_t here so $u_t = 0$. Using this algorithm, we iteratively update the distribution of $X_t \sim \mathcal{N}(\boldsymbol{\mu_t}, \boldsymbol{\Sigma_t})$

according to Z_t using the updating rule described in [Thrun et al. 2005]. This model has an additional error when sound source changes its velocity, due to the linear velocity assumption.

5 Auditory steering

In order to integrate sound localization into a predictive steering framework, we must first be able to estimate the collision boundary of a sound emitting agent, and its current velocity. This allows an agent to build a complete spatial representation of all obstacles in the environment that it sees or hears, allowing it to exploit both sensory modalities for goal-directed collision avoidance.

5.1 Sound Obstacle

As we have seen, X_t corresponds to a normal distribution $\mathcal{N}(\boldsymbol{\mu_t}, \boldsymbol{\Sigma_t})$, where $\boldsymbol{\mu_t}$ represents the continuous estimate of the sound source location (velocity related terms are not used here), and $\boldsymbol{\Sigma_t}$ represents the spatial uncertainty. We introduce the concept of *sound obstacle*, whose position corresponds to $\boldsymbol{\mu_t}$ and size corresponds to $\boldsymbol{\Sigma_t}$, which intuitively means that we are choosing a "core" area of the Gaussian distribution.

$$P(\boldsymbol{X}) = \frac{1}{2\pi|\boldsymbol{\Sigma_t}|} e^{-\frac{1}{2}(\boldsymbol{X}-\boldsymbol{\mu_t})^T \boldsymbol{\Sigma_t}(\boldsymbol{X}-\boldsymbol{\mu_t})} \geq threshold$$

$$(\boldsymbol{X}-\boldsymbol{\mu_t})^T \boldsymbol{\Sigma_t}(\boldsymbol{X}-\boldsymbol{\mu_t}) \leq threshold, \boldsymbol{X} = \left(\begin{array}{c} x \\ y \end{array}\right)$$

The exact solution of this inequality will lead to a quadric which defines the boundary of the sound obstacle. We do not care about the exact shape because human relies on prior knowledge of sound source size, which can be denoted as $\boldsymbol{\Sigma_P}$. Based on prior semantic knowledge of sound type, e.g. the sound is emanated from a car or a human, we could add that information to the predicted shape of the sound obstacle, and thus the final space occupancy of sound obstacle can be $\boldsymbol{\Sigma_P} + \boldsymbol{\Sigma_t}$. A conservative agent might choose a larger size of sound obstacles. Since we do not care about the exact shape of the region, for computational simplicity of the steering algorithm, we could use a sphere with radius $\sqrt{\sigma_x \sigma_y}$, or more conservatively $max\{\sigma_x, \sigma_y\}$.

5.2 Velocity Estimate

The direct way to estimate the velocity of a sound obstacle is to calculate the gradient of position, however, we do not need to do so in our model. In the framework of Kalman Filter, it is quite easy – we already add velocity in the state space of Kalman Filter, so the output of Kalman Filter already contains velocity. Recall that $X_t \sim \mathcal{N}(\boldsymbol{\mu_t}, \boldsymbol{\Sigma_t})$, so smoothed velocity estimate is contained in $\boldsymbol{\mu_t}$ and velocity uncertainty is contained in $\boldsymbol{\Sigma_t}$.

5.3 Predictive Collision Avoidance using Sound Obstacles

Agents compute sound obstacle and predicted velocity of the sound emitting object based on the sound localization result. This information can be easily fed into traditional synthetic vision-based steering methods [Ondřej et al. 2010] to incorporate hearing into collision avoidance. In our framework, we exploit the estimated velocity of the sound obstacle for predictive collision avoidance.

Each agent keeps a list of its neighbors, obtained by vision or via sound localization. Notice that each agent might localize the same sound source at different positions due to localization error. For vision, we model visibility as a foveal cone to limit the number

of obstacles an agent can see. We treat sound obstacles as traditional velocity obstacles in the HRVO framework [Snape et al. 2011]. HRVO is designed to accommodate sensor noise for collision avoidance in robotics, and effectively handles the inaccuracy in the predicted position, velocity, and collision boundary of the sound obstacles.

6 Results

Sound Localization. Figure 9 illustrates the localization and tracking of one or more sources in the absence of obstacles, with high localization accuracy due to the absence of audio distortion.
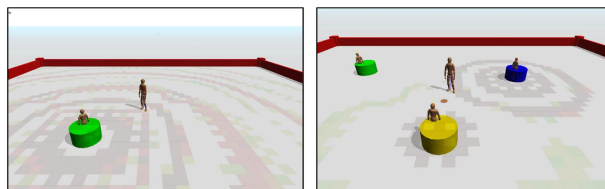

Figure 9: *a) Tracking a single source. b) Tracking multiple sources.*
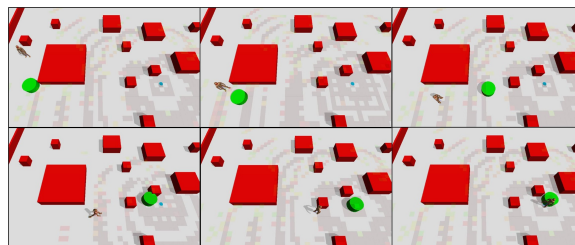

Figure 10: *Source finding, also illustrating the influence of obstacles on localization. Agent goes to the estimated position of sound (green) each step; finally, the agent gets to the exact source position (blue marker).*

Our algorithm localizes the source near the corner of the obstacle, along the shortest path from source to receiver. This makes sense because in this case, the nearest *sub-source* or secondary source is around the corner, so our algorithm outputs the sub-source instead of the source. More generally, for a complex obstacle arrangement, our algorithm will point to the nearest sub-source or average of several nearest sources. However, if the receiver is trying to find the position of source using our algorithm, as the agent is approaching current sub-source, the sub-source will ultimately converge to the real source position, as shown in Figure 10. Notice also how obstacles influence the localization process.

Navigating to a Sound Source. The predicted position of a sound emitting object can be used as a target to navigate an agent towards it. This can be used to produce chase simulations where an agent can exploit both vision and hearing to chase other agents. Figure 11 illustrates this example, also shown in the accompanying video.

Avoiding Collision with Sound-Emitting Objects. Figure 1 illustrates a simple example where blind agents are crossing a highway

Figure 11: *The agent is using a vision-sound multi-model steering to chase the target agent. When he cannot see target anent, sound provides clues for navigation.*

with bi-directional traffic. Agents cannot see the cars, but predictively avoid collisions by hearing them, and are able to cross the highway safely.

Blind Corner. Agents use hearing to predict the locations of potential crossing threats around corners, as illustrated in Figure 12.
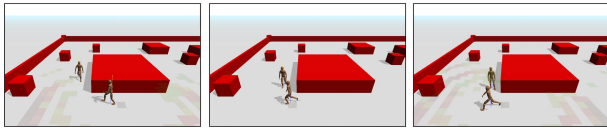

Figure 12: *Corner case. One agent hears that another agent is approaching from the other side of the blind corner and stops.*

6.1 Computational Performance

Our experiment is setup in Unity using the ADAPT character animation platform [Shoulson et al. 2013]. Auditory simulation is implemented as a C++ plugin and localization is implemented in C#, on a Core i-7 dual-core MacBook. The computational cost of sound simulation is proportional to the number of sound sources (denoted as s), and the cost of sound localization is proportional to the number of sound-receiver pairs (denoted as p). Figure 13 shows the computation cost with increasing values of s and p.
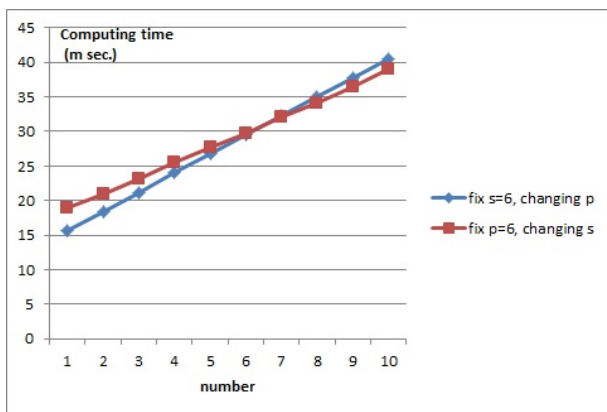

Figure 13: *Performance: computing time per update for a) fixed source number (s=6) and varying source-receiver pairs (p from 1 to 10). b) fixed source-receiver pair number (p=6) and varying source number (s from 1 to 10).*

Phase	Avg. time per update (ms)
sound simulation	2.25
source-receiver pair	2.76

Table 1: *Average time per update.*

7 Conclusions and Future work

In this paper, we discussed the process of simulating autonomous virtual agents capable of hearing and localizing sounds in the environment, and using this information for audio-driven steering. We describe a variety of cases that demonstrate the benefits of integrating hearing into traditional vision-only agent models. Currently, auditory perception is limited to steering and collision avoidance, without speech and communication. In our model, only an energy value is contained in the sound packet, which can convey more information such as semantic message, an segment of record or

even computer generated sound. There are two possible approaches: the first is that speech signals are contained in the sound packets and propagating in the virtual world, and the agent processes perceived signals using pattern recognition and natural speech processing techniques, which also provide a human computer interface with speech and make it possible that human directly communicates with virtual agent. Another approach is that only semantic information is contained in the packets and signal processing part is skipped, making the simulation more efficient.

One straightforward approach to model human response to sound might be directly mimicking the human auditory and perception system. However, this would require high sound simulation accuracy, making it very costly even to simulate the sound perception for already one agent, effectively precluding the simulation of a large amount of agents. Instead, this paper introduces many simplifications that allow us to simulate large amounts of agents in real-time. In order to acquire high accuracy, key auditory properties such as ITD, IID and HRTF (head related transfer functions) need to be properly modeled in future work. Other potential improvements include representing the detector subject as a circular normal distribution and using a wrapped Kalman Filter for azimuthal source tracking [Traa and Smaragdis 2013].

References

ALLEN, J. B., AND BERKLEY, D. A. 1979. Image method for efficiently simulating small-room acoustics. *The Journal of the Acoustical Society of America 65*, 943.

ANTONACCI, F., FOCO, M., SARTI, A., AND TUBARO, S. 2004. Real time modeling of acoustic propagation in complex environments. In *Proceedings of 7th International Conference on Digital Audio Effects*, 274–279.

BONNEEL, N., DRETTAKIS, G., TSINGOS, N., VIAUD-DELMON, I., AND JAMES, D. 2008. Fast modal sounds with scalable frequency-domain synthesis. In *ACM Transactions on Graphics (TOG)*, vol. 27, ACM, 24.

CISKOWSKI, R. D., AND BREBBIA, C. A. 1991. *Boundary element methods in acoustics*. Computational Mechanics Publications Southampton, Boston.

DISSANAYAKE, M. G., NEWMAN, P., CLARK, S., DURRANT-WHYTE, H. F., AND CSORBA, M. 2001. A solution to the simultaneous localization and map building (slam) problem. *Robotics and Automation, IEEE Transactions on 17*, 3, 229–241.

FUNKHOUSER, T., CARLBOM, I., ELKO, G., PINGALI, G., SONDHI, M., AND WEST, J. 1998. A beam tracing approach to acoustic modeling for interactive virtual environments. In *Pro-*

ceedings of the 25th annual conference on Computer graphics and interactive techniques, ACM, 21–32.

FUNKHOUSER, T., TSINGOS, N., CARLBOM, I., ELKO, G., SONDHI, M., WEST, J. E., PINGALI, G., MIN, P., AND NGAN, A. 2004. A beam tracing method for interactive architectural acoustics. *The Journal of the Acoustical Society of America 115*, 739.

HELBING, D., AND MOLNAR, P. 1995. Social force model for pedestrian dynamics. *PHYSICAL REVIEW E 51*, 42–82.

HUANG, J., OHNISHI, N., AND SUGIE, N. 1997. Sound localization in reverberant environment based on the model of the precedence effect. *Instrumentation and Measurement, IEEE Transactions on 46*, 4 (aug), 842 –846.

HUANG, P., KAPADIA, M., AND BADLER, N. I. 2013. SPREAD: Sound Propagation and Perception for Autonomous Agents in Dynamic Environments. In *ACM SIGGRAPH/EUROGRAPHICS SCA*.

IHLENBURG, F. 1998. *Finite element analysis of acoustic scattering*, vol. 132. Springer.

JAMES, D. L., BARBIČ, J., AND PAI, D. K. 2006. Precomputed acoustic transfer: output-sensitive, accurate sound generation for geometrically complex vibration sources. In *ACM TOG*, vol. 25, 987–995.

KAGAWA, Y., TSUCHIYA, T., FUJII, B., AND FUJIOKA, K. 1998. Discrete huygens' model approach to sound wave propagation. *Journal of Sound and Vibration 218*, 3, 419 – 444.

KAPADIA, M., AND BADLER, N. I. 2013. Navigation and steering for autonomous virtual humans. *Wiley Interdisciplinary Reviews: Cognitive Science*, n/a–n/a.

KAPADIA, M., SINGH, S., HEWLETT, W., AND FALOUTSOS, P. 2009. Egocentric Affordance Fields in Pedestrian Steering. In *Interactive 3D graphics and games*, ACM, I3D '09, 215–223.

KAPADIA, M., BEACCO, A., GARCIA, F., REDDY, V., PELECHANO, N., AND BADLER, N. I. 2013. Multi-domain real-time planning in dynamic environments. In *ACM SIGGRAPH/Eurographics SCA*, 115–124.

KRISTIANSEN, U., AND VIGGEN. 2010. Computational methods in acoustics. DEPARTMENT OF ELECTRONICS AND TELECOMMUNICATIONS, NTNU.

LITOVSKY, R. Y., COLBURN, H. S., YOST, W. A., AND GUZMAN, S. J., 1999. The precedence effect.

LOOMIS, J., KLATZKY, R., PHILBECK, J., AND GOLLEDGE, R. 1998. Assessing auditory distance perception using perceptually directed action. *Attention, Perception, Psychophysics 60*, 966–980. 10.3758/BF03211932.

MARTIN, K. D. 1995. A computational model of spatial hearing. Thesis (M.S.) Massachusetts Institute of Technology. Dept. of Electrical Engineering and Computer Science.

MEHRA, R., RAGHUVANSHI, N., SAVIOJA, L., LIN, M. C., AND MANOCHA, D. 2012. An efficient gpu-based time domain solver for the acoustic wave equation. *Applied Acoustics 73*, 2, 83 – 94.

MONZANI, J.-S., AND THALMANN, D. 2000. A sound propagation model for interagents communication. In *Virtual Worlds*, Springer, 135–146.

O'BRIEN, J. F., SHEN, C., AND GATCHALIAN, C. M. 2002. Synthesizing sounds from rigid-body simulations. In *ACM SIGGRAPH/Eurographics SCA*, 175–181.

ONDŘEJ, J., PETTRÉ, J., OLIVIER, A.-H., AND DONIKIAN, S. 2010. A synthetic-vision based steering approach for crowd simulation. *ACM Trans. Graph. 29*, 4 (July), 123:1–123:9.

PARIS, S., PETTR, J., AND DONIKIAN, S. 2007. Pedestrian reactive navigation for crowd simulation: a predictive approach. *Computer Graphics Forum 26*, 3, 665–674.

PELECHANO, N., ALLBECK, J. M., AND BADLER, N. I. 2007. Controlling individual agents in high-density crowd simulation. In *ACM SIGGRAPH/Eurographics SCA*, 99–108.

PELECHANO, N., ALLBECK, J. M., AND BADLER, N. I. 2008. *Virtual Crowds: Methods, Simulation, and Control*. Synthesis Lectures on Computer Graphics and Animation.

RAGHUVANSHI, N., NARAIN, R., AND LIN, M. C. 2009. Efficient and accurate sound propagation using adaptive rectangular decomposition. *Visualization and Computer Graphics, IEEE Transactions on 15*, 5, 789–801.

RAGHUVANSHI, N., SNYDER, J., MEHRA, R., LIN, M., AND GOVINDARAJU, N. 2010. Precomputed wave simulation for real-time sound propagation of dynamic sources in complex scenes. *ACM Transactions on Graphics (TOG) 29*, 4, 68.

SHAO, W., AND TERZOPOULOS, D. 2005. Autonomous pedestrians. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, ACM, 19–28.

SHOULSON, A., MARSHAK, N., KAPADIA, M., AND BADLER, N. I. 2013. Adapt: the agent development and prototyping testbed. In *ACM SIGGRAPH I3D*, 9–18.

SINGH, S., KAPADIA, M., HEWLETT, B., REINMAN, G., AND FALOUTSOS, P. 2011. A modular framework for adaptive agent-based steering. In *ACM SIGGRAPH I3D*, 141–150 PAGE@9.

SINGH, S., KAPADIA, M., REINMAN, G., AND FALOUTSOS, P. 2011. Footstep navigation for dynamic crowds. In *ACM SIGGRAPH I3D*, 203–203.

SNAPE, J., VAN DEN BERG, J., GUY, S. J., AND MANOCHA, D. 2011. The hybrid reciprocal velocity obstacle. *Robotics, IEEE Transactions on 27*, 4, 696–706.

STRUMILLO, P. 2011. *Advances in Sound Localization*. InTech.

THALMANN, D., AND MUSSE, S. R. 2013. *Crowd Simulation, Second Edition*. Springer.

THRUN, S., BURGARD, W., FOX, D., ET AL. 2005. *Probabilistic robotics*, vol. 1. MIT press Cambridge, MA.

TRAA, J., AND SMARAGDIS, P. 2013. A wrapped kalman filter for azimuthal speaker tracking.

VAN DEN BERG, J., LIN, M., AND MANOCHA, D. 2008. Reciprocal velocity obstacles for real-time multi-agent navigation. In *ICRA*, IEEE, 1928–1935.

WILKIE, D., VAN DEN BERG, J., AND MANOCHA, D. 2009. Generalized velocity obstacles. In *IROS*, IEEE, 5573–5578.

YU, Q., AND TERZOPOULOS, D. 2007. A decision network framework for the behavioral animation of virtual humans. In *ACM SIGGRAPH SCA*, Eurographics Association, 119–128.