OPTIMIZING COVERAGE OF THREE-DIMENSIONAL WIRELESS SENSOR NETWORKS BY MEANS OF PHOTON MAPPING

Bruce Johnson
Jason Isaacs

Hairong Qi

Naval Surface Warfare Center,
Panama City Division
Panama City, FL 32407, USA

Department of Electrical Engineering and Computer Science
University of Tennessee
Knoxville, TN 37996, USA

ABSTRACT

As wireless sensor networks applied to 3D spaces gain in prominence, it becomes necessary to develop means of understanding how to optimize 3D sensor coverage while taking into account the environmental conditions in which they operate. To accomplish this goal, this paper presents the Sensor Placement Optimization via Queries (SPOQ) simulation algorithm. It determines where to place the minimal number of simulated bistatic sensors such that they cover as much of the single-source-illuminated virtual environment as possible. SPOQ performs virtual sensor placement optimization by means of making queries to the photon map generated by the photon mapping algorithm and uses this query output as input to a prevailing modified sensor placement algorithm. Since SPOQ uses photon mapping, SPOQ can take into account static or dynamic simulated environmental conditions and can use exploratory or precomputed sensing. The SPOQ method is computationally efficient, requiring less memory than other sensor placement solutions.

1 INTRODUCTION

Wireless sensor networks (WSNs) applied to 3D spaces are gaining in importance for WSN researchers seeking to understand how to perform sensing in (typically) non-terrestrial environments. Applications are being created that assume that the WSN is deployed in the air or underwater where the constituent sensor nodes may occupy different heights or depths, respectively. For example, small aerial wireless sensor nodes have been developed that are meant to perform fire monitoring (Purohit et al. 2011); underwater sensor nodes have been developed for the purpose of performing pollution monitoring (Khan and Jenkins 2008).

Quantifying the ability of a WSN to sense the environment it is deployed in is known as the *coverage problem* (Huang and Tseng 2005). This problem seeks to answer the question: "How well do the sensors monitor the space they occupy?" This question can be expressed as *k-coverage* (Huang and Tseng 2005) which asks "Given an integer k and a sensed region R, can it be assured that each point in R is monitored by k sensors?" Providing knowledge of an environment by a WSN enhances the ability of those operating the WSN to more fully understand what events are taking place in that environment and the subsequent actions that should be taken in response to those events. Lack of information (i.e. inadequate sensor coverage) could conceal vital information that would compromise the ability of the WSN operator to make an intelligent assessment of their environment. Therefore, the importance of the coverage problem (especially when it is applied to 3D spaces) cannot be understated.

The coverage problem has been considered by the mathematics community in the subject area known as *visibility optimization* (Tsai et al. 2004). One such visibility optimization problem is the art gallery problem (AGP) that derives its name from the hypothetical situation wherein an art gallery owner seeks to

have every painting covered with as few sensors as possible. The AGP is similar to *k*-coverage except that it seeks to use the smallest *k* sensors necessary to assure that every point in *R* is covered by at least one sensor. (For the purposes of this paper, we will consider only the AGP.) The AGP is usually stated in the following manner: "What is the minimum number of sensors necessary to assure that every point in the monitored space is sensed?" (Tsai et al. 2004). Since the AGP is known to be NP-hard, solutions to the AGP are suboptimal (Goroshin, Huynh, and Zhou 2011).

The hidden assumption behind the formulation of the AGP is that all points in *R* that fall within the sensing range of the sensor are equally capable of being monitored from a given observation point. While this level of mathematical abstraction of an ideal environment and its sensor is useful, it does not take into account mitigating factors that can disrupt the ability of a sensor to obtain optimal coverage. Such factors include:

1. Environments that are less than ideal and are prone to fog, dust, rain and so forth;
2. Environments wherein the ability of a given sensor to sense its surroundings changes with time as a monitored space transitions from day to nighttime; and
3. Determination of how to solve the AGP in 3D environments that are not known prior to the WSN's deployment because the monitored environment has not been explored yet or has been altered after the passage of a unit of time.

In order to be able to address these factors, we will examine the nature of the photon. The photon is the fundamental quantum constituent of electromagnetic (EM) radiation. Since most sensors are meant to detect a range of frequencies in the EM spectrum, the photon may also be regarded as being responsible for transferring information from a point in the monitored space to the sensor that detects them. When a sensor node is regarded as being responsible for detecting the energy transmitted by the photons reflecting off of a surface in an environment, the AGP can be recast as the following question: "What is the minimum number of sensors necessary to assure that every photon available to convey information about the monitored space is sensed?" Solving this formulation of the AGP relies upon the creation of a model describing how a set of simulated photons will propagate in a 3D virtual environment. Furthermore, the photon model requires an efficient data structure with small memory requirements and a fast search time.

Modeling the behavior of photons is a well-established area of study in the graphics community. One such algorithm that produces both a photon propagation model and data structure that allows us to solve the AGP is *photon mapping* (Jensen 2001). The photon mapping algorithm provides the following:

1. The presumption that virtual photons are propagated in a 3D virtual environment;
2. The use of a balanced kd-tree data structure (Cormen, 2009) (known as a *photon map*) requiring $O(\log p)$ access time and $O(p)$ memory to store the results propagating *p* photons;
3. The ability to take into account how participating media affects the behavior of photons that propagate within the 3D virtual environment; and
4. The ability to alter the number of photons that will propagate through the virtual environment.

The contribution our paper makes is showing how the use of the photon mapping algorithm, in combination with a modified prevailing AGP-solving algorithm, allows us to address the three less-than-ideal factors mentioned above, i.e., environments prone to interference, changed sensing capabilities and being unknown prior to or after the passage of time.

2   RELATED WORK

Consideration of how to solve the coverage problem has been the subject of much research. These efforts can be generally divided into two categories. The first category is dedicated to producing an algorithm meant to establish and maintaining coverage and the second category is dedicated to a theoretical analysis of coverage performance.

Algorithms have been developed to maintain sensor coverage while using the minimum number of sensors (Xing et al. 2005; Yan, He, and Stankovic 2003). These efforts presume that the sensing range is a circle and can sense something within their sensing range with certainty. Other efforts to maintain sensing coverage make the same circular sensing range assumption but have a probabilistic sensing ability (Ahmed 2005, Ren 2007). Theoretical studies regarding how to solve the *k*-coverage problem have been formulated for different sensor deployment strategies (Kumar, Lai, and Balogh 2004; Wan and Yi 2006). Another study (Xing et al. 2009) continues to presume that sensor has a circular sensing range, but inject probability-based uncertainty in the sensor's ability to provide coverage.

Most articles published about obtaining sensor coverage presume that the virtual environment to be sensed is 2D. There have been a few papers (Alam and Haas 2006, Alam and Haas 2008) published which presume that the environment to be sensed is 3D. These works presume that the sensor is surrounded by some space-filling volume which represents the limits of their omnidirectional sensor's range. Their aim is to place the sensors in the 3D virtual environment in such a way that fills up as much space as possible while using as few sensors as possible.

A key feature of the previous efforts is the construction of a *sensor grid* where the virtual sensors will be placed. In the 2D and 3D cases, the sensor grid is created by placing a tessellated set of points or filling a volume with points, respectively, such that a sensor positioned at point *x* is displaced from its neighbors by some distance. In both the 2D and 3D cases, the object (whose dimensions are defined by *surfaces*) occupying the environment to be sensed has no sensor grid point placed inside the object's boundaries.

In the mathematics community, sensor coverage is given the name visibility coverage (Tsai et al. 2004). Whatever is covered within the sensing range of a sensor is regarded as being *visible*. The sensor is regarded as an *observer* and the position the observer resides is an *observer station*. The sensor grid mentioned above is thus called an *observer grid*. Determining what is visible by an observer is accomplished by tracing a ray from the observer station until it intersects with an object. The point of intersection on the object is thus regarded as being visible and is thus *observed*.

Let O be a set of points that comprise a surface and let it be a member of the compact subset Ω of $\mathfrak{R}^d$ where $\mathfrak{R}^d$ is a *d*-dimensional subset of the real numbers. The set O is an *occluder* residing in *environment* Ω. The set $X = \Omega \setminus O$ is the set of points in which an omnidirectional observer may be stationed and $x_o \in X$ is some observer station. An observer grid $G \subseteq X$ comprises of a sequence of points $g_0, g_1, \ldots g_n$ wherein element g_i is separated by its neighboring elements by some distance such that the distance does not exceed the boundaries of G. This distance assumption ensures that at least one observer station is found within the confines of G.

The AGP is posited by the mathematics community as the following question:

- What is the minimal number of observers necessary to ensure that the maximum number of points in Ω are observed?

Tsai et al. (2004) formulated a method for achieving visibility in a 3D virtual environment without considering optimal observer placement. Building upon this work, Cheng and Tsai (2005) solved the AGP by dismissing global optimality and considered only local maxima as a suitable solution. This is due to the fact that the AGP is known to be non-convex and thus gradient ascent methods are incapable of determining globally optimized visibility coverage (Goroshin, Huynh, and Zhou 2011). Goroshin, Huynh, and Zhou (2011) expanded upon the work of Cheng and Tsai (2005) by providing a greedy iterative

algorithm within the level set framework in order to solve the AGP. The efforts made by Cheng and Tsai (2005) and Goroshin, Huynh, and Zhou (2011) have centered on using the so-called "fast visibility sweeping" algorithm as a means of performing "implicit ray tracing" (Goroshin, Huynh, and Zhou 2011) in order to provide the information necessary for solving the AGP in 2D and 3D environments.

Given environment Ω, a *known* environment allows the observers to have already sensed Ω prior to solving the AGP. An *unknown* environment means that observers cannot have sensed Ω prior to solving the AGP and further implies that they will have to explore *X* or *G* in order to solve the AGP.

The level set methods advanced in Tsai et al. (2004), Goroshin, Huynh, and Zhou (2011), and Cheng and Tsai (2005) are useful in a known environment. The computation cost of the AGP-solving algorithms presented in Tsai et al. (2004), Goroshin, Huynh, and Zhou (2011), and Cheng and Tsai (2005) requires $O(lm)$ steps to perform their visibility optimization computation where *l* is the number of sensors and *m* is the number of observation points in the virtual environment. However, in order to perform the visibility optimization calculation provided by Goroshin, Huynh, and Zhou (2011), the visibility of every single observer point must be precomputed and stored at a cost of $O(m^{2d})$ memory where *m* is the number of elements contained in the environment in consideration and *d* is the dimension of the environment.

The basic assumption made by these coverage algorithms is that the sensor nodes which comprise the WSN contain a transceiver on the sensor node's platform. Our work regards the sensor nodes as being passive receiver-only i.e. bistatic platforms. (Modeling of the establishment of coverage for bistatic sensors, is not well-established.) The coverage algorithms presented in Tsai et al. (2004), Goroshin, Huynh, and Zhou (2011), and Cheng and Tsai (2005) presume that the sensors are monostatic – i.e. sensors that contain both the sensor and the emitter on the same platform.

3   PHOTON MAPPING

The idea of photon mapping rests upon the decoupling of the energy used to illuminate an environment from the illuminated scene's geometry. The photon mapping algorithm creates a queryable data structure known as the *photon map* whose data structure is that of a balanced kd-tree (Cormen 2009) which stores the location (among other properties) of the simulated photons that have been cast into the virtual environment by a photon source. The photon mapping algorithm is a two pass algorithm wherein the first pass entails the construction of a photon map and the second pass entails the rendering of the virtual environment by means of ray-tracing using the information contained in the photon map.

3.1   Pass 1: Photon Emission and Propagation

The first pass of the photon mapping algorithm consists of two steps: photon emission and photon propagation. The first step entails creating *p* photons and then launching them into the scene. Each of the p photons represents a fragment of the total energy of the photon source. The initial direction of the *p* photons is based upon the type of photon source used – spherical, rectangular or directional.

For the second step, the propagation or *bouncing* of the photon is measured. Once a photon touches an object in the virtual environment, it will be either absorbed, transmitted or reflected. The determination as to whether a photon is absorbed, transmitted or reflected is accomplished by means of the *Russian roulette* Monte Carlo technique (Avro and Kirk 1990). If the photon is absorbed, then no further bounces are done. If the photon hits a transmissive surface (such as a pane of glass), the photon's resulting bounce-direction is determined by Snell's Law. If the photon hits a specular-reflective surface (such as polished metal), the photon bounces in another direction based upon a calculation of the bidirectional radiance diffuse function (BRDF) (He et al. 1991). If the photon hits a diffusely-reflective surface (such as a wooden desk), the photon's power, final point of intersection on a surface and direction of entry are stored in the photon map and is bounced in another direction based upon a calculation of the BRDF. The interactions of different types of photons on a virtual environment is illustrated in Figure 1.
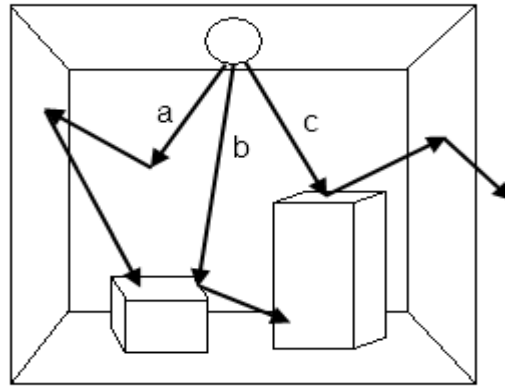
Figure 1: Illustration of photon bouncing in a simple scene. Photon **a** is diffusely reflected until it is absorbed, photon **b** is specularly reflected and photon **c** is diffusely reflected until it leaves the environment. Note the presence of the photon source in the upper center of the environment.

The Russian roulette technique eliminates unimportant photons. If a photon is absorbed, then no further computational resources need be expended on modeling the behavior of that photon. Let $\zeta \in [0, 1]$ be a uniformly distributed random variable and let t, d and s be the material's transmission coefficient and its diffuse and specular reflection coefficients, respectively. The photon is stored in the photon map if $\zeta \in [0, d]$, specularly reflected if $\zeta \in (d, s + d]$, transmitted if $\zeta \in (s + d, s + d + t]$, and absorbed if $\zeta \in (s + d + t, 1]$. The accommodation of photons that lie outside the bandwidth of visible light may be accomplished by adjusting the probabilities of absorption, reflection and transmission.

If we want to take into account the effects of participating media such as fog, rain, smoke and so forth, Jensen advocates the use of a *volume photon map* which stores the photons' interaction with the media (Jensen 2001). Adjustments to the propagation of the photons through the media are made based upon the media's density, whether it is hetero- or homogeneous and whether or not the photon's scattering by the media is anisotropic. The photons which escape interaction with the participating media are stored in a global photon map per the rules explained above. The concept of participating media as it applies to photon mapping is illustrated in Figure 2.
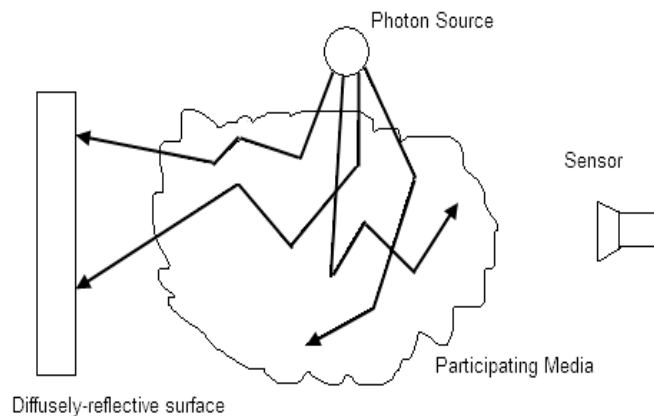

Figure 2: Photon scattering due to the participating media in the center of the figure is stored in a volume photon map whereas those photons which strike the diffusely-reflective surface on the left are stored in the global photon map. Photons in both maps can be perceived by the sensor on the right.

3.2 Pass 2: Using the Photon Map

The photons contained in the photon map have conventionally been used in the second pass of the photon mapping algorithm by a ray-tracer in order to provide a rendering of a 3D virtual environment. Rendering is achieved by determining the total amount of energy radiated from every surface in the virtual environment which has been intersected by a ray cast by a ray-tracer at some observation point. A query to the photon map yields those photons that contribute to the radiant energy - expressed as a color - found at that point of intersection. A query takes the shape of a sphere (or *query-sphere*) of radius *r* centered at point x_o (or *query-point*). For our purposes, the query-sphere and the query-point are analogous to a sensor node's sensing range and location, respectively. Since the photon map utilized by the photon mapping algorithm is guaranteed to be balanced, the computational cost of getting the photons that populate the query sphere is $O(\log p)$ where *p* is the total number of photons in the virtual environment.

The query-point and the rays point of intersection on the radiating surface are traditionally regarded as equivalent. But we claim that we are not limited by this presupposition. The query-sphere can also be regarded as a means of gathering the ambient energy and therefore information emanating from surfaces in the environment to be sensed. If a photon stored within the photon map is located a distance which is less than or equal to *r*, that virtual photon is returned by the photon map. In Section 4 we will discuss how we decide if the returned photon is indeed visible to the sensor.

By making a sensor's sensing range to be the equivalent in dimension of a query-sphere, we presume that sensors are omnidirectional. This presumption limits the types of sensors that we can simulate. This limitation is overcome by the fact that if we are given a query-sphere, we can carve it into a viewing frustum or any other sort of shape. Once the query-sphere has been reshaped into a *query-volume*, we can make the query-volume directional by only allowing the inclusion of those photons that have arrived from a particular direction into the query-volume. For example, if we carve a query-sphere into a viewing frustum, we can make the viewing frustum only perceive those photons that have arrived from the front of the viewing frustum and exclude the rest.

By using the photon mapping algorithm, we are now enabled to solve the AGP in a 3D environment while taking into consideration the three factors we considered in Section 1 namely, sensor attenuation, altered sensing capabilities due to environmental effects and sensing an unknown environment prior to or after the passage of time. The photons are all assumed to be propagating in a 3D environment. By using a volumetric photon map, we may take into account participating media and by controlling the number of photons launched into a virtual environment we can simulate a photon-rich or photon-poor environment. Since the photon map's data structure's information may be accessed quickly and its memory requirements are scalable, we can quickly evaluate an unknown sensor's sensing ability.

4 MEASURING COVERAGE

Recall that the photon map stores the photon's power, final point of intersection on a surface and direction of entry. In our application, the stored photons are modified in their data structure in that they store the surface's normal vector translated to point of the photon's intersection. We will show how this variable aids in the determination of the coverage of a monitored space.

Distinguishing between those photons that are responsible for contributing to the sensor's ability to sense and those that don't is a well-established problem in photon mapping and has been subject to consideration previously (Jensen 2001). We use the name *photon culling* to describe the act of testing and removing those photons found within the query-sphere which do not contribute to sensor coverage. Those photons which are occluded by an intervening obstacle are said to be *shadowed*.

A means of culling non-contributing photons is to perform a dot product test on the normal vector emanating from point of intersection of the photon and the query-point. If the dot product performed on the query-point and the normal emanating from the point upon which the photon resides is greater than 0,

then the photon is pointed to the query-point and is thus visible to the sensor. This technique for photon culling is similar to what was proposed by Jensen (2001).

Given x representing a point on the surface that has been illuminated by the photon mapping algorithm and an observer residing at point x_o in the virtual environment, a photon map query supplemented with the photon culling method described above produces the following function $f{:}R{\rightarrow}D$ such that range $R = \{(x, x_o) \mid x, x_o \in \Re^n$ and $x_o \neq x \}$ and domain $D = \{\Re^+$ if x is pointing to x_o, 0 if a photon has been completely absorbed at x, $\Re^-$ if x is pointing away from $x_0\}$ where $\Re^+$ and $\Re^-$ are the sets of positive and negative real numbers, respectively.

Having this function available leads to the production of the following level set function (Goroshin, Huynh, and Zhou 2011):

$$\Phi(x; x_0) = \min_{z \in L(x,x_0)} \Psi(z)$$

where Φ is non-positive at x when x is invisible to the observer at x_o and positive otherwise, $L(x, x_o)$ is a line segment connecting the observer at x_o to the point x and $\Psi > 0$ when z is an unobstructed line segment connecting x and x_o. This function permits the distinction between the surfaces which are and are not visible to an observer. In our terminology, the function distinguishes between those photons (and the surfaces they illuminate) which are (in)visible to an observer.

5   COMBINING CONCEPTS

The establishment of visibility is a consequence of determining the amount of non-occluded space encompassed within the viewing area seen by n observers. This concept is expressed in the following equation inspired by Goroshin, Huynh, and Zhou (2011)

$$V(g_1, g_2, \dots g_n) = \int_G H\big(\Phi(y; g_i)\big)dy$$

where $H(\bullet)$ is the one-dimensional Heaviside step function, with $H(0) = 0$ and G is the set of observer grid positions comprised of elements $g_1, g_2, \dots g_n$. The entire possible visible volume is

$$\int_G H\big(\Psi(y)\big)dy.$$

The normalized visible volume, V_{norm}, describes the fraction of coverage proved by the observers and is provided by the following equation

$$V_{norm} = \frac{\int_G \big(\sum_{i=1}^n H\big(\Phi(y; x_i)\big)\big)dy}{\int_G H\big(\Psi(y)\big)dy} \tag{1}$$

and $0 \leq V_{norm} \leq 1$.

The method for establishing observer placement in order to achieve maximum coverage is provided by the following equation

$$\underset{g_f \in G}{\operatorname{argmax}} \int_G H\left(H\big(\Phi(y; g_j)\big) - H\left(\sum_{\substack{i=1 \\ i \neq j}}^n H\big(\Phi(y; g_i)\big)\right)\right)dy. \tag{2}$$

The first term gives the visible region provided by the jth observer and the second term is the visible volume provided by all other observers.

The equation provides an exhaustive search of the optimal location to place one observer at a time. The process is repeated for each observer in a manner similar to simulated annealing. According to Goroshin, Huynh, and Zhou (2011) "the optimal observer positions may not be unique; therefore an optimal position is chosen at random in order to avoid limit cycles. Observers are updated in random order for the same reason."

The pseudocode for the Sensor Placement Optimization via Queries (SPOQ) simulation algorithm responsible for solving the AGP in a *known* environment Ω is inspired by Goroshin, Huynh, and Zhou (2011) and is given below.

Algorithm 1 SPOQ

---
1. Construct a photon map
2. **for** $g_i \in G$ do
3. Store the p photons observed by g_i
4. **end for**
5. Initialize V_{norm}^k using Equation (1)
6. Initialize $V_{norm}^{k-1} = 0$
7. **while** $V_{norm}^k > V_{norm}^{k-1}$ **do**
8.   **for** all n observers **do**
9.   choose an observer j at random without replacement
10.   find a globally optimal solution to Equation (2) for observer j
11.   **end for**
12. $V_{norm}^{k-1} = V_{norm}^k$
13.   Recompute V_{norm}^k using new observer positions
14. **end while**

What distinguishes this version of SPOQ from Goroshin, Huynh, and Zhou (2011) is that at each grid point $g_i \in G$, the number of photons obtained at that grid point are stored prior to executing the while loop. In an *unexplored* environment, then steps $1 - 4$ would be omitted and step 10 would entail making a query to the photon map rather than doing a look up of what had been stored previously at grid point g_i.

We use the following algorithm detailing the steps taken in step 10 of SPOQ:

Algorithm 2 SPOQ: Photon query step

---
1. Set observer j's visibility to 0
2. Get the current visibility volume for all observers other than j
3. **for** all $g_i \in G$ **do**
4.   Search the photon map to get photons who contribute to the visibility of observer j using photon culling
5.   Set the visible volume equal to the summation of all previously seen visible volumes
6. **end for**
7. Select maximum visible volume

Recall that determining which photons contribute to visibility is a matter of culling non-contributing photons via the dot-product test described in Section 4.

5.1   Algorithm Analysis

By using queries to the photon mapping algorithm's photon map as input to SPOQ, it then becomes possible to solve the AGP problems while taking into account participating media in complex 3D virtual environments illuminated by different types of EM radiation. The computation and memory requirements

depend on the number of photon sources used and whether the environment is known or unknown. When dealing with an unknown environment, for each of the l sensors who will occupy the m observation points, queries to the photon map with photon culling have a computational cost of O($q•\log q•\log p$) where p is the number of photons launched into the virtual environment, q is the number of photons returned by a query and $q < p$. The visibility optimization in an unknown environment requires O($l•m•q\log q•\log p$) operations whereas a known environment requires O($l•m$) operations where $l << m$. The difference stems from the fact that in an unknown environment a new photon map query with photon culling must be performed at each observer position in order to update the information obtained as the observer explores the 3D virtual environment whereas in a known environment the photon map query result may be precomputed at each observation point and then utilized by each respective sensor. The memory requirement of SPOQ when used in an unknown environment is O(p) and O($l•m•q$) for a known environment.

6   RESULTS

The SPOQ method may be used by following the steps given below:

1. Create the virtual environment's model.
2. Apply photon mapping to the virtual environment.
3. Construct the observer grid.
4. Apply SPOQ.

Step 1 entails using some modeling program. This program may be similar to Google's Sketchup (Sketchup 2013) which offers many free, well-crafted models through its 3D Warehouse (3D Warehouse 2013). Constructing the observer grid in step 2 is a matter of determining the dimensions of the observer grid and then populating it with points separated by some regular predetermined distance such that they do not exceed the dimensions of the observer grid. These steps are illustrated in Figure 3.
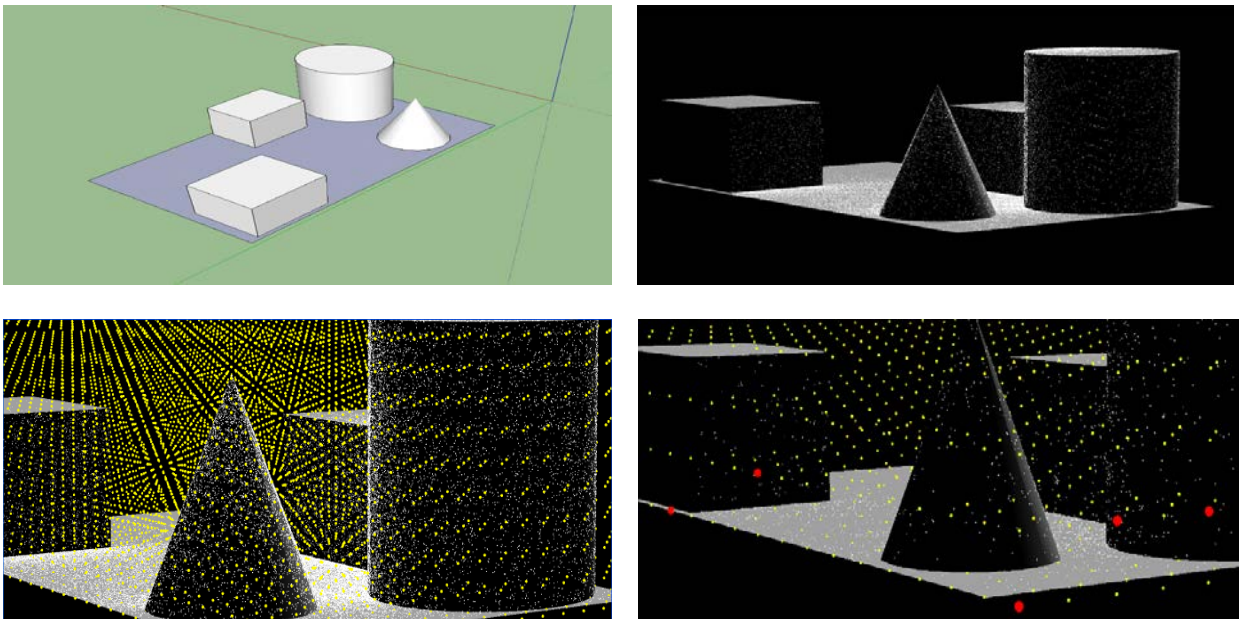

Figure 3: The four steps taken to perform SPOQ.

Tables 1, 2, and 3 represent the coverage obtained by the SPOQ simulation algorithm on the model depicted in Figure 3 using successively greater numbers of photons. The successively greater number of

photons represents an environment that transitions from a photon poor or "dark" environment to a photon rich or "bright" environment. The environment was unknown prior to using SPOQ.

Table 1: Coverage obtained by SPOQ using 1000 photons on a 268-polygon model and 1188 grid points.

Observers	% Photons Covered	Time (seconds)
2	95.3	0.8
3	96.5	4.5
4	95.8	3.9
5	97	2.9

Table 2: Coverage obtained by SPOQ using 5000 photons on a 268-polygon model and 1188 grid points.

Observers	% Photons Covered	Time (seconds)
2	95.2	4.7
3	96.5	14.1
4	96.6	9.4
5	97.3	35.3

Table 3: Coverage obtained by SPOQ using 10000 photons on a 268-polygon model and 1188 grid points.

Observers	%Photons Covered	Time(seconds)
2	93.1	25.2
3	96.2	16.1
4	96.9	14.2
5	97.1	17.8

7   CONCLUSIONS AND FUTURE WORK

This work represents the first step in the development of a means of obtaining optimal sensor coverage in multiple types of 3D environments. There are three areas of development that we want to pursue, namely

1. the accommodation of multiple heterogeneous sensing ranges;
2. the accommodation of sensors which use sound rather than EM spectra; and
3. the accommodation of real time updates to the photon map in order to represent a changing environment.

The first effort may be accomplished by the use of an area of study in mathematics known as *sphere packing* (Hales 1992). This problem seeks to find an arrangement of spheres - that may have unequal radii - such that the spheres fill as much of the volume as possible. In the context of SPOQ, this sphere arrangement represents an observer grid containing sensors with heterogeneous sensing ranges. The application of SPOQ would proceed in the manner mentioned above except for the fact that it would be highly unlikely that a particular sensor occupying a given observation point would have a sensing range equal to that of its neighbor.

The second effort has been considered before by the introduction of *phonon mapping* (DeGreve 2006). The photon mapping algorithm's construction of the photon map need not be confined strictly to the EM spectrum. It has been shown that by extending the concept of the photon to be any discrete packet of information-bearing energy propagating through a medium, the behavior of sound traversing through air can also be modeled. This packet of sound energy is known as a *phonon* rather than a photon. While the phonon's operational characteristics in air have been considered previously, the transmission of a phonon through water has not. We expect that it would entail the use of the equivalent of a volumetric photon map except that it would be applied to water rather than air. Providing coverage for maritime environments using phonons is an "over the horizon" area of research that has not been fully developed.

The third effort may be accomplished by means of performing parallel computation on a photon map in order to update it in real time. This topic has been subject to consideration previously (Zhou 2008). Essentially, what these efforts are aiming to achieve is the rapid updating to a kd-tree in response to the movement of objects within the virtual environment. Each photon can operate independently of every other photon and can be assigned its own thread on an *n*-core machine. Once the kd-tree is updated on some core in an *n*-core machine, SPOQ can be used on that particular core. Given an *n*-core processor, SPOQ could provide the optimal sensor placement of a dynamic scene operating *n* time steps into the future. It should be noted that while parallelizing the photon map algorithm is well established, parallelizing the phonon mapping algorithm has not been done.

We look forward to further developing SPOQ in order to make it a more powerful and accommodating AGP solver.

ACKNOWLEDGMENTS

REFERENCES

Ahmed, N., S. S. Kanhere, and S. Jha. 2005. "Probabilistic Coverage in Wireless Sensor Networks." In *Proc. of the 30th IEEE Conf. on Local Computer Networks,* 673-681.Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Avro, J., and D. B. Kirk. 1990. "Particle Transport and Image Synthesis Computer Graphics." In *SIGGRAPH '90*: *Proc. of 17th Conf. on Computer Graphics and Interactive Techniques*, 63-66. New York: ACM Press.

Cheng, L-T. and Y-H. Tsai, 2005. "Visibility Optimization Using Variational Approaches." *Communications in Mathematical Sciences* 3(3): 425-451.

Cormen, T. H., C. E. Leiserson, R. L. Rivest, and C. Stein. 2009. *Introduction to Algorithms. 3rd ed.* Cambridge, MA:MIT Press.

De Greve, B., T. Willems, T. De Muer, and D. Botteldooren. 2006. "The Effect of Diffuse Reflections on Noise Maps Using Phonon Mapping." In *ICSV '13*: *Proc. of the 13th International Congress on Sound and Vibration*.

Goroshin, R., Q. Huynh, and H. M. Zhou. 2011. "Approximate Solutions to Several Visibility Optimization Problems." *Communications in Mathematical Sciences* 9(2): 535-550.

Hales, T. C. 1992. "The Sphere Packing Problem." *Journal of Computational and Applied Mathematics* 44 (1): 41 – 76.

He, X., K. Torrance, F. Sillon, and D. A. Greenberg. 1991. "Comprehensive Physical Model for Light Reflection." In *SIGGRAPH '91*: *Proc. of 18th Conf. on Computer Graphics and Interactive Techniques*, 175-186. New York: ACM Press.

Huang, C-F. and Y-C. Tseng. 2005. "The Coverage Problem in a Wireless Sensor Network." *Mobile Networks and Applications* 10(4): 519 - 528.

Jensen, H. W. 2001. *Realistic Image Synthesis Using Photon Mapping*. New York: A. K. Peters.

Kajiya, J. T. 1986. "The Rendering Equation." In *SIGGRAPH '86*: *Proc. of 13th Conf. on Computer Graphics and Interactive Techniques*, 143-150. New York: ACM Press.

Khan, A. and L. Jenkins. 2008. "Undersea Wireless Sensor Network for Ocean Pollution Prevention." In *COMSWARE '08*: *Proc. of3rd Int'l Conf. on Communication Systems Software and Middleware and Workshops*, 2-8. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Kumar, S., T. H. Lai, and J. Balogh. 2004. "On *k*-coverage in a Mostly Sleeping Sensor Network." In *MobiCom '04*: *Proc. of 10th Int'l Conf. on Mobile Computing and Networking,*144-158. New York: ACM Press.

Nazrul Alam, S. M. and Z. J. Haas. 2006. "Coverage and Connectivity in Three-Dimensional Networks." In *MobiCom '06*: *Proc. of 12th Int'l Conf. on Mobile Computing and Networking,*346-357. New York: ACM Press.

Nazrul Alam, S. M. and Z. J. Haas. 2008. "Coverage and Connectivity in Three-Dimensional Underwater Sensor Networks." *Wireless Communication and Mobile Computing*, 8(8), 995-1009.

Purohit, A, P., Z. Sun, F. Mokaya, and P. Zhang. 2011. "SensorFly: Controlled-Mobile Sensing Platform for Indoor Emergency Response Applications." In *ISPN'11*: *Proc. of the 10th Int'l Conf. on Information Processing in Sensor Networks,* 223-234. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Ren, S., Q. Li, H. Wang, X. Chen, and X. Zhang. 2007. "Design and Analysis of Sensing Scheduling Algorithms Under Partial Coverage for Object Detection in Sensor Networks." *IEEE Transactions of Parallel Distributed Systems* 18(3): 334-350.

Sketchup 2013. Accessed April 5. http://www.google.com/sketchup.

Tsai, Y-H., L-T. Cheng, S. Osher, P. Burchard, and G. Sapiro. 2004. "Visibility and Its Dynamics in a PDE Based Implicit Framework." *Journal of Computational Physics* 199(1):260-290.

Xing, G., X. Wang, Y. Zhang, C. Lu, R. Pless, and C. Gill. 2005. "Integrated Coverage and Connectivity Configuration for Energy Conservation in Sensor Networks." *ACM Transactions on Sensor Networks* 1(1): 36-72.

Xing, G., R. Rui, B. Liu, J. Wang, X. Jia, and C-W. Yi. 2009. "Data Fusion Improves the Coverage of Wireless Sensor Networks." In *MobiCom '09*: *Proc. of 15th Int'l Conf. on Mobile Computing and Networking,*157-168. New York:ACM Press.

Yan, T., T. He, and J. A. Stankovic. 2003. "Differentiated Surveillance for Sensor Networks." In *SenSys '03*: *Proc. of 1st Int'l Conf. on Embedded Networked Sensor Systems,* 51-62. New York: ACM Press

Zhou, K., Q. Hou, R. Wang, and B. Guo. 2008. "Real-Time KD-tree Construction on Graphics Hardware." *ACM Transactions on Graphics* 27(5): Article 126.

3D Warehouse 2013. Accessed April 5. http://sketchup.google.com/3dwarehouse.

AUTHOR BIOGRAPHIES

BRUCE JOHNSON is pursuing a PhD from the University of Tennessee in Electrical Engineering and is a research engineer for the Naval Surface Warfare Center in Panama City, Florida. He is interested in wireless network sensor simulation and numerical analysis. His email is bruce.a.johnson5@navy.mil.

HAIRONG QI received the B.S. and M.S. degrees in computer science from Northern JiaoTong University, Beijing, China in 1992 and 1995 respectively, and the Ph.D. degree in computer engineering from North Carolina State University, Raleigh, in 1999. She is currently a Professor with the Department of Electrical Engineering and Computer Science at the University of Tennessee, Knoxville. Her current research interests are in advanced imaging and collaborative processing in resource-constrained distributed environment, hyperspectral image analysis, and bioinformatics. Her email is hqi@utk.edu.

JASON ISAACS is a research engineer in the Science, Technology, Analysis, and Simulation Division at the Naval Surface Warfare Center (NSWCPCD) in Panama City, FL since 2008 after graduating with a Ph.D in electrical engineering from the Florida State University. His research is focused on the use of machine learning, computer vision, and image processing for automatic target recognition applications. Specific areas of interest are in representational learning, knowledge transfer, and graph-based methods for shape analysis. His email is jason.c.isaacs1@navy.mil.