

Article

## Ranging in an Underwater Medium with Multiple Isogradient Sound Speed Profile Layers

Hamid Ramezani \* and Geert Leus

Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, 2826 CD Delft, The Netherlands; E-Mail: g.j.t.leus@tudelft.nl

\* Author to whom correspondence should be addressed; E-Mail: h.mashhadiramezani@tudelft.nl; Tel.: +31-1-5278-6280; Fax: +31-1-5278-6190.

Received: 17 December 2011; in revised form: 20 February 2012 / Accepted: 24 February 2012 / Published: 2 March 2012

---

**Abstract:** In this paper, we analyze the problem of acoustic ranging between sensor nodes in an underwater environment. The underwater medium is assumed to be composed of multiple isogradient sound speed profile (SSP) layers where in each layer the sound speed is linearly related to the depth. Furthermore, each sensor node is able to measure its depth and can exchange this information with other nodes. Under these assumptions, we first show how the problem of underwater localization can be converted to the traditional range-based terrestrial localization problem when the depth information of the nodes is known a priori. Second, we relate the pair-wise time of flight (ToF) measurements between the nodes to their positions. Next, based on this relation, we propose a novel ranging algorithm for an underwater medium. The proposed ranging algorithm considers reflections from the seabed and sea surface. We will show that even without any reflections, the transmitted signal may travel through more than one path between two given nodes. The proposed algorithm analyzes them and selects the fastest one (first arrival path) based on the measured ToF and the nodes' depth measurements. Finally, in order to evaluate the performance of the proposed algorithm we run several simulations and compare the results with other existing algorithms.

**Keywords:** localization; ranging; sound speed profile; ray tracing; underwater acoustic sensor networks

---

## 1. Introduction

Available wireless sensor network (WSN) localization techniques rely on mutual distances between sensors [1], which are for instance estimated from time of flight (ToF) measurements. In homogeneous medium, like air, where the propagating wave speed is constant, the mutual distances between the nodes are linearly related to the ToFs. In contrast, the propagating wave speed inside an inhomogeneous medium is not constant, and depends on the location. In such a medium, the ToF between two nodes depends not only on the sound speed profile (SSP) of that medium but also on the position of the two nodes [2]. Therefore, the ToF is not linearly proportional to the Euclidean distance between the nodes, and the distance-dependent localization techniques are not appropriate for inhomogeneous media. As a result, they should be modified to ToF-based techniques.

In [3], the problem of localizing a node in an underwater environment with a known depth-dependent SSP is considered. As the target node measures the ToF from an anchor node, the corresponding constant range interval surface for the measured ToF is constructed. To construct a single constant range interval surface, the path trajectory for each departing ray from an anchor node is calculated. Then on each path trajectory, a point is selected based on the measured ToF. All these points together form the constant range interval surface. After sufficient ToF measurements are taken, the position of the target node is estimated as a point whose sum of squared distances from all these surfaces is minimum. The main drawback of this approach is the computational complexity which is dependent on the network size and the required accuracy.

In our earlier work [4], we consider the problem of localizing a target node in an underwater environment with an isogradient SSP. There, we directly work with the ToF measurements, and we localize a target node based on the ToF equations. Since the algorithm is based on an analytical approach, the computational complexity is acceptable. Although the assumption of a single isogradient SSP is appropriate for a deep underwater medium, it is not valid for the entire environment. The sound speed at a given point in an underwater medium is affected by the salinity, water temperature and pressure of that point [5], and in general this causes the SSP to vary nonlinearly and even non-monotonically with respect to (w.r.t.) the depth, especially in a shallow underwater medium.

In [6] and [7], the localization of an underwater WSN is investigated, in a 3D environment assuming knowledge of the nodes' depth. These algorithms can help us to convert the inhomogeneous underwater localization problem into a traditional 2D homogeneous distance-dependent localization problem, which is well-studied in the literature [8,9]. We will show that as the depth information is known a priori, the ToF between two nodes only depends on the horizontal distance and thus also on the distance between the nodes. Hence, having the ToF measurement between two nodes, the pair-wise distance can be computed. As an example, based on the depth information and the SSP, a look up table (LUT) is built in [10], which relates the ToF measurement to the horizontal distance between two nodes. The algorithm of [10] is very fast, but to scan the whole inhomogeneous environment, a huge LUT is required which may not be practical. Furthermore, the SSP in an underwater medium is subject to changes in temperature and conductivity, and any change in SSP degrades the LUT accuracy and upsets the localization performance.

In [11], the problem of ranging in an inhomogeneous underwater environment is considered. A numerical range estimator is proposed which is based on reconstructing the slanted path using Fermat's

principle and calculus of variations. Basically, after depth and time measurements, the authors form an integral equality which is taken over the depth between nodes. Then, they try to numerically calculate the constant defined by Snell's law. Afterwards, by using the computed constant, they calculate the horizontal distance between the nodes through another integral equality. Their work is really comprehensive since with any given SSP, the horizontal distance is computable, but their algorithm to compute the constant value (defined by Snell's law) has an ambiguity, because in an inhomogeneous medium it is common that a traveling ray from one node to another node passes a given depth more than once. Since the depth of a point on a traveling ray trajectory is not a monotonic function of the depth, this phenomenon casts an ambiguity on the value of any integral taken w.r.t. the depth along the traveling path.

In this work, we analyze the acoustic signal propagation between two sensor nodes in an underwater environment. We use a ray-tracing approach to model the propagation which is a valid approximation for high-frequency signal transmission [10]. We assume that the underwater medium is composed of different layers with an isogradient SSP, which is a practical model for the actual SSP of the entire environment [12,13]. We will show that in such an environment, if the depth information of two nodes located on a specific ray is known, then the positions of the crossing points, where the ray trajectory and the layer boundaries meet each other, can be obtained through a set of polynomial root finding equations. Based on these equations we are able to distinguish among different possible transmission paths between the nodes, and determine the fastest one. The proposed method for finding the fastest transmission path between the nodes can handle reflections from the surface and the seabed by adding more polynomial equations to the set. Another contribution of this paper is a novel method for accurate ranging between the nodes. The proposed algorithm computes the horizontal distance between two nodes based on the ToF and depth measurements. The algorithm estimates the range of a target by minimizing the difference between the measured ToF and the constructed ToF estimated from the known map in an iterative manner.

The rest of the paper is organized as follows. We describe the network model in Section 2, and we compute the ToF *versus* the node positions in Section 3. Next, in Section 4, we propose our ranging algorithm, and we extract its CRB. Then in Section 5, we evaluate the performance of the proposed algorithm through several simulations. Finally, we conclude the paper in Section 6.

## 2. Network Model

Consider  $K$  anchor nodes with known locations and one target node in an underwater acoustic sensor network (UASN). The goal of the system is to estimate the position of the target node with ToF and depth measurements. To relate the wave ToF to the node position inside an underwater medium, we are faced with the classical problem of how an individual ray behaves in the medium, and how a ray departing one node arrives at the other node.

Assume that the wave speed in a Cartesian coordinate system is a function of the position and is defined by  $c(x, y, z)$ . Then, we can compute the ToF between two nodes,  $\mathbf{x}^S$  and  $\mathbf{x}^E$  as

$$t = \int_{s(\mathbf{x}^S, \mathbf{x}^E)} \frac{ds}{c(x, y, z)} \quad (1)$$

where  $s$  is the arc length, which is related to the ray path according to the standard ray equation

$$\frac{d}{ds} \left( \frac{1}{c(x, y, z)} \frac{d\mathbf{x}}{ds} \right) = -\frac{1}{c^2(x, y, z)} \nabla c(x, y, z) \quad (2)$$

with  $\mathbf{x} = \mathbf{x}(s) = [x(s), y(s), z(s)]^T$  the position on the ray determined by the arc length  $s$ . It is clear from Equation (1) that in an inhomogeneous medium the ToF between two nodes is not linearly dependent on the distance between them, and is a function of the two node positions. Using the ToF measurements to  $K$  anchors, the location of a node can be obtained as,

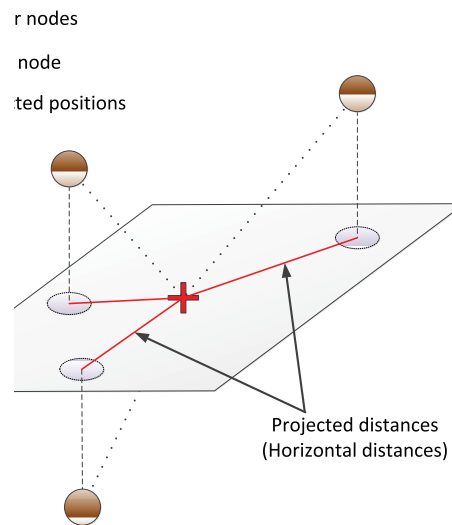
$$\hat{\mathbf{x}} = \min_{\mathbf{x}} |\mathbf{t}(\mathbf{x}) - \mathbf{t}_m|^2 \quad (3)$$

where  $\mathbf{t}(\mathbf{x})$  is a  $K \times 1$  vector with the  $k$ -th element representing the ToF between the target node and the  $k$ -th anchor node, and  $\mathbf{t}_m$  is a vector denoting the noisy ToF measurements to all  $K$  anchors. To localize a node in a 3D inhomogeneous environment without depth information at least four anchors are needed, and the localization process is known as quadrilateration. If the depth information is available, it is possible to localize a node with just three anchors, but still we have to work with ToFs. For more information about how the anchor nodes inside the network are selected and how they can communicate with each other, the reader is referred to [6].

In an underwater medium where the sound speed varies only with depth, the availability of depth information does not only allow us to work with only three anchors, it also opens the door to convert the time-based localization problem into a traditional range-based one as explained next.

Since the SSP is only a function of depth, the problem of ray tracing between two nodes has a cylindrical symmetry around the line parallel to the  $z$ -axis and passing through one of the nodes. Hence, we can map the problem of ray tracing into a vertical plane that crosses the two nodes. As the depth of each node inside the plane is known, the ToF only depends on the horizontal range between the nodes. In other words, the horizontal distance between the nodes is the only variable that determines the ToF. Suppose now that the ToF of the first arrival path as a function of the horizontal distance is an invertible function, meaning that the ToF of the first arrival path between two points located at specific depths is a monotonic (actually increasing) function of the horizontal distance between the nodes. Then, one parameter can be computed from the other, and we are basically able to estimate the distance between the nodes using the corresponding ToFs. If the above assumption does not hold for a given environment, then there would be ambiguities in the ranging problem. However, this monotonicity is generally observed.

The conversion of a 3D underwater localization problem to a 2D one can now be explained as depicted in Figure 1. Using the ToFs and depth information, the unknown node computes the pair-wise distances to the other nodes. Then it projects the estimated distances on a horizontal plane at its depth. Finally, based on the projected distances, 2D multilateration is performed to compute its position [7]. Therefore, localizing a node in this case only requires the knowledge of the projected distances to the anchors or the estimated horizontal distances.

**Figure 1.** Projection of pair-wise distances on the horizontal plane crossing the target.

### 3. ToF Versus Node Positions

In order to relate the ToF to the node positions, we first require to find which ray departing the source reaches a specific destination. In this section, we analytically find the rays that can travel between two nodes with known positions, and we compute their corresponding ToFs based on their trajectories. It is worth mentioning that in an underwater medium with fixed SSP, each ray departing a source can be uniquely characterized by its departing angle.

Here the SSP is considered as a piece-wise linear function of the depth which is a valid approximation according to the measured data [14]:

$$c^{(j)}(z) = a^{(j)}z + b^{(j)}, z^{(j-1)} < z < z^{(j)}, j \in 1, \dots, N \quad (4)$$

where  $z$  represents the depth,  $a^{(j)}$  and  $b^{(j)}$  are related to the chemical and physical characteristics of the  $j$ -th isogradient SSP layer, and  $N$  is the number of layers. In our previous work [4], we show how a ray can travel between two nodes located inside an isogradient SSP underwater environment. We review this work first for completeness.

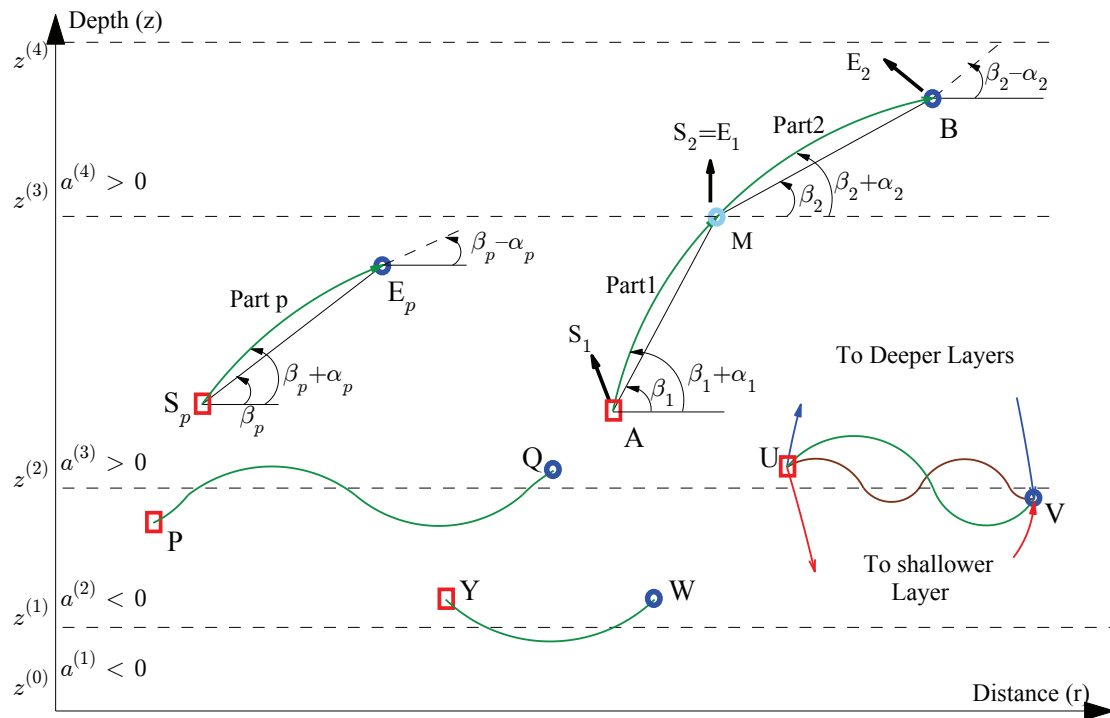
#### 3.1. ToF Versus Node Positions in a Single Layer

##### Exact Propagation

In a single layer, each truncated ray (indexed by  $p$ ) between two points, *i.e.*,  $S_p$  and  $E_p$  in Figure 2, can be uniquely characterized if the position of the starting point, position of the end point, and SSP are known. In order to simplify the notation, we index the SSP from now on by the truncated ray index instead of the layer index. For instance, for the  $p$ -th truncated ray located at the  $j$ -th layer, we introduce the new notation

$$c_p(z) = a_p z + b_p = c^{(j)}(z) = a^{(j)}z + b^{(j)} \quad (5)$$

Figure 2. Samples of ray trajectories as they travel through different layers.



The relation between the ToF and the node positions can then be extracted from a set of differential equations characterized by Snell’s law [10],

$$\frac{\cos \theta}{c_p(z)} = \frac{\cos \theta_p^S}{c_p(z_p^S)} = \frac{\cos \theta_p^E}{c_p(z_p^E)} = k_0, \text{ and } \theta \in \left( -\frac{\pi}{2}, \frac{\pi}{2} \right) \tag{6}$$

where  $\theta_p^S$  and  $\theta_p^E$  are the ray angles at the starting and end points, respectively,  $z_p^S$  and  $z_p^E$  represent the depth of the starting and end node, respectively, and  $k_0$  is constant along a ray traveling between the nodes (see Figure 2). Moreover, the parameters  $\theta$  and  $z$  represent the angle and depth of a given point along the ray. Now, we can write

$$\partial r = \frac{\partial z}{\tan \theta} \tag{7a}$$

$$\partial s = \frac{\partial z}{\sin \theta} \tag{7b}$$

$$\partial t = \frac{\partial s}{c_p(z)} \tag{7c}$$

where  $s$  is the arc length of a ray traveling between the two nodes, and  $t$  is its corresponding travel time. From Equations (5) and (6), by taking derivatives w.r.t.  $z$  and  $\theta$ , we can write

$$\partial z = -\frac{1}{a_p k_0} \sin \theta \partial \theta \tag{8}$$

Using the above differential equations, for the two points we have [4]

$$r_p^E - r_p^S = \sqrt{(x_p^E - x_p^S)^2 + (y_p^E - y_p^S)^2}, \quad (9a)$$

$$X_p = \frac{z_p^E - z_p^S}{r_p^E - r_p^S} \quad (9b)$$

$$Y_p = \begin{cases} \frac{K_p}{X_p}, & z_p^S \neq z_p^E \\ \frac{0.5a_p(r_p^E - r_p^S)}{b_p + a_p z_p^S}, & z_p^S = z_p^E \end{cases} \quad (9c)$$

$$K_p = \frac{0.5a_p(z_p^E - z_p^S)}{b_p + 0.5a_p(z_p^E + z_p^S)}, \quad (9d)$$

$$\tan \beta_p = X_p \quad (9e)$$

$$\tan \alpha_p = Y_p \quad (9f)$$

$$\theta_p^S = \beta_p + \alpha_p \quad (9g)$$

$$\theta_p^E = \beta_p - \alpha_p \quad (9h)$$

$$t_p = \frac{-1}{a_p} \left[ \ln \frac{1 + \sin \theta_p^E}{\cos \theta_p^E} - \ln \frac{1 + \sin \theta_p^S}{\cos \theta_p^S} \right] \quad (9i)$$

where  $r_p^E - r_p^S$  is the horizontal distance between the points,  $\mathbf{x}_p^S = [x_p^S, y_p^S, z_p^S]^T$  and  $\mathbf{x}_p^E = [x_p^E, y_p^E, z_p^E]^T$  are the coordinates of the starting and end points, respectively, and  $t_p$  is the traveling time of a truncated ray between these two nodes. Note that  $\beta_p$  represents the angle of the straight line connecting the nodes w.r.t. the horizontal axis, and  $\alpha_p$  is the angle between the actual ray and this straight line. From Equations (9g) and (9h), it can be seen that the ray deviations from the straight line at the starting and end point are the same but have opposite signs.

### Linear ToF Approximation

The simple linear ToF approximation based on the depth information between two nodes in the same layer, which assumes a straight-line ray propagation, can be derived as

$$t_{\text{app},p} = \int_{z_p^S}^{z_p^E} \frac{1}{\sin(\beta_p) c_p(z)} dz = \frac{d^{S_p E_p}}{z_p^E - z_p^S} \frac{1}{a_p} \ln \left( \frac{c_p^E}{c_p^S} \right) = \frac{d^{S_p E_p}}{c_p^E - c_p^S} \ln \left( \frac{c_p^E}{c_p^S} \right) \quad (10)$$

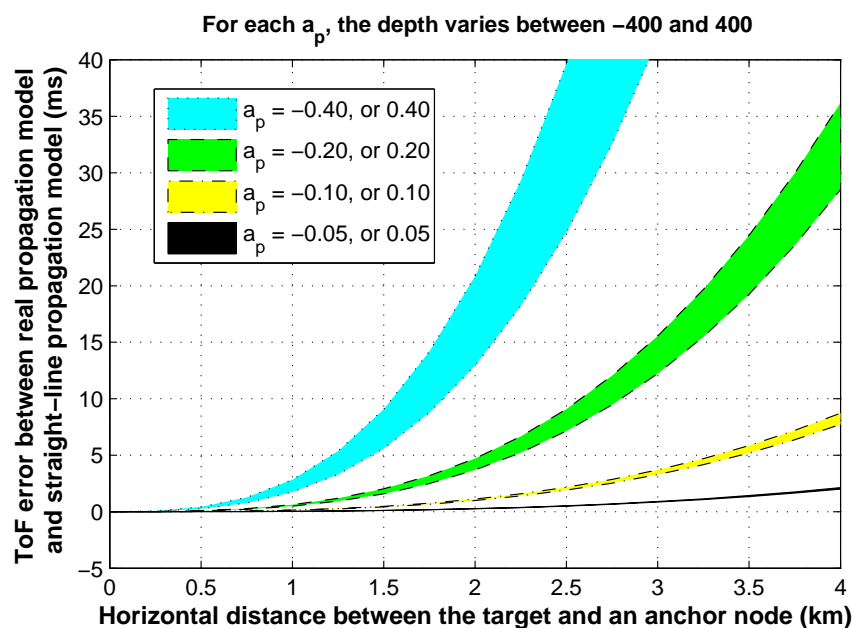
where  $d^{S_p E_p}$  is the straight-line distance between the starting point and end point of the truncated ray, and  $c_p^S$  and  $c_p^E$  are the sound speeds at the starting and end point, respectively. From Equation (10), it can be seen that  $t_{\text{app},p}$  has a linear dependency on the range which is analogous to homogeneous medium. A similar approximation for a multi-layer medium which assumes a straight-line wave propagation between two nodes,  $\mathbf{x}^S$  and  $\mathbf{x}^E$ , can be obtained as

$$t_{\text{app}} = \frac{d^{SE}}{z^E - z^S} \sum_{p=1}^P \frac{1}{a_p} \ln \left( \frac{c_p^E}{c_p^S} \right) \quad (11)$$

where  $d^{\text{SE}}$  is the distance between the nodes, and  $P$  is the number of single-layer parts of the straight ray. For a straight ray, the number of single layer parts between two nodes is exactly the same as the number of layers it passes. The accuracy of the straight-line approximation will be evaluated in the numerical section.

Fermat's principal, which also leads to Snell's law, states that the path traveled by a ray between two points is the path that can be traversed in the least amount of time. Therefore, the approximated time based on a straight-line ray propagation,  $t_{\text{app}}$ , is always greater than the actual ToF. For instance, in Figure 3, it is shown that the difference between  $t_{\text{app},p}$  and the actual ToF,  $t_p$ , is always positive. Here, we assume that the sound speed at  $z = 0$  is  $b_p = 1,480$  m/s, and we compute the ToF error for different values of depth, distance, and sound speed steepness. It is shown that as the absolute value of the SSP steepness,  $|a_p|$ , increases this error grows exponentially. In addition, for the low values of SSP steepness, the change in depth has less effect on this error in comparison with the change in pair-wise distance.

**Figure 3.** ToF error of the straight-line propagation model in a single layer for different values of range and depth.



### 3.2. ToF Versus Node Positions for Two Adjacent Layers

We start our analysis by considering a ray traveling between two points in adjacent layers. As illustrated in Figure 2, there are many ways for a ray to travel from U to V. An analysis of all possible rays is feasible, and we will later on discuss this. Now, we only focus on a ray which crosses the intermediate boundary only once, and does not propagate into other layers except these two adjacent layers such as the ray from A to B in Figure 2. In this scenario, when the  $r$  coordinate of the crossing point, M, is computed, we are able to relate the positions of the two nodes to the ToF. It can be seen that the ray has two parts, one indexed by  $p = 1$ , and the other by  $p + 1 = 2$ . The ending point of the first part is the starting point of the second part. Thus, the two parts of the ray can be related to each other according to

$$\theta_2^S = \theta_1^E \quad (12)$$



Another representation for Equation (12) can be obtained by taking the tangent from both sides of the equation. Using Equations (9e) to (9h), the boundary equation can then be modified to

$$\frac{X_2 + Y_2}{1 - K_2} = \frac{X_1 - Y_1}{1 + K_1} \quad (13)$$

For a two-part ray, the combination of Sub-Equations (9b), (9c), and (9d) for each part, together with boundary Equation (13) forms a third-order polynomial root finding problem where the roots represent the possible  $r$  coordinates of the crossing point M. Notice that the node positions and the depth of M are known and as a result, the parameters  $K_1$  and  $K_2$  can be computed easily,  $X_1$  ( $X_2$ ) is inversely related to  $Y_1$  ( $Y_2$ ), and the only unknown parameter is  $r^M$  which determines  $X_1$  and  $X_2$ . Since there are at most three roots for a third order polynomial, there are at most three ways for a ray departing at A to reach B (note that we are still assuming that a ray crosses the boundary once, and propagates only in these two adjacent layers). For each of these possible rays the ToF can be calculated as,

$$t = \frac{-1}{a_1} \left[ \ln \frac{1 + \sin \theta_1^E}{\cos \theta_1^E} - \ln \frac{1 + \sin \theta_1^S}{\cos \theta_1^S} \right] + \frac{-1}{a_2} \left[ \ln \frac{1 + \sin \theta_2^E}{\cos \theta_2^E} - \ln \frac{1 + \sin \theta_2^S}{\cos \theta_2^S} \right] \quad (14)$$

### 3.3. Pattern Definition for Multi-Layer Ray Propagation

To simplify the multi-layer analysis, we define the concept of *ray pattern*. A ray pattern is a set consisting of all possible rays that can travel between two points. For example, a ray pattern of 2.1.1.2 means that, the ray departs the starting point from the second layer, goes to the first layer, hits the surface, and arrives to the second node in the second layer. Therefore, a *ray pattern* has several properties. First, the number of digits used in the *ray pattern* indicates the number of single-layer parts a ray consists of. Second, it shows in which layer each part of a ray is located. Third, the reflection from the sea surface and the seabed can easily be modeled by this concept. Using the *ray pattern* concept, we are able to show how a ray can travel in a given medium, and which pattern may host the fastest ray.

### 3.4. ToF Versus Node Positions According to a Given Pattern

The procedure of ToF computation, as a function of the node positions for a ray which has multiple single-layer parts is the same as the two-part ray, but with more boundary equations. The combination of all these equations may form a higher order polynomial root finding problem, which consequently may increase the number of ways that a ray can travel between the two points. To predict how a ray may travel inside a multi-layer underwater area we introduce several lemmas bellow.

**Lemma 1:** The sound speed in a layer with zero steepness SSP is constant. Thus, the wave propagation inside that layer is along a straight line, and for that reason the parameter  $\alpha_p$  for each truncated ray at that layer is zero. In this case, the relation between the ToF and the node positions is modified into

$$t_p = \frac{1}{b_p} \sqrt{(r_p^E - r_p^S)^2 + (z_p^E - z_p^S)^2} \quad (15)$$

**Lemma 2:** Rays are bent toward the region where the sound speed is lower.

**Proof:** If we place Equation (9b) into Equation (9c), for two nodes at different depths we obtain

$$Y_p = \frac{0.5a_p(r_p^E - r_p^S)}{b_p + 0.5a_p(z_p^E + z_p^S)} \quad (16)$$

The denominator of Equation (16) is the average of the sound speed at the two points and is always positive, and hence using Equation (9f) we understand that the angle  $\alpha_p$  has the same sign as  $a_p$ . When  $a_p$  is larger than zero, the ray angle at the starting point is greater than the ray angle at the end point or any other point along the ray trajectory. Therefore, the ray angle has a tendency to become smaller and to bend up where the sound speed is lower. On the other hand, when  $a_p$  is smaller than zero, the ray angle has a tendency to become larger and to bend down where again the sound speed is lower. Indeed, as depicted in Figure 2, within a layer, a ray bends toward the region where the sound speed is lower.

**Lemma 3:** In a layer, the depth along a truncated ray between two points,  $S_p$  and  $E_p$ , can exceed the region  $[z_p^S, z_p^E]$  if and only if (iff) the sign of  $\theta_p^S \theta_p^E$  is negative. The excess value can be computed as,

$$\Delta z = \begin{cases} \frac{a_p z_p^S + b_p}{a_p \cos \theta_p^S} (1 - \cos \theta_p^S), & \text{if } |\theta_p^S| < |\theta_p^E| \\ \frac{a_p z_p^E + b_p}{a_p \cos \theta_p^E} (1 - \cos \theta_p^E), & \text{if } |\theta_p^S| > |\theta_p^E| \end{cases} \quad (17)$$

**Proof:** In a single layer, the depth of a node along a given part as a function of the angle can be derived from Equation (6)

$$z = \frac{1}{a_p} \left( \frac{\cos \theta}{k_0} - b_p \right) \quad (18)$$

where  $k_0 = \frac{\cos \theta_p^S}{a_p z_p^S + b_p} = \frac{\cos \theta_p^E}{a_p z_p^E + b_p}$  is a positive constant for a given ray. It is obvious that  $z$  follows the behavior of  $\cos \theta$ , and its extremum occurs when  $\theta$  is zero. In other words, the depth of a node on a truncated ray exceeds the region  $[z_p^S, z_p^E]$  iff the signs of the angles at the starting and end points differ from each other. As a result, the excess value can be computed as,

$$\Delta z = \begin{cases} \max_{\theta} z(\theta) - \max \{z_p^S, z_p^E\} & \text{if } a_p > 0 \\ \min_{\theta} z(\theta) - \min \{z_p^S, z_p^E\} & \text{if } a_p < 0 \end{cases} \quad (19)$$

which leads to Equation (17).

In this way, it can be understood that if  $\Delta z$  for a one-part ray in a single layer is so large that a ray part crosses another layer, the assumption of a one-part ray propagation has to be changed into a three-part ray propagation, and the equations have to be reorganized accordingly (see for instance the ray in Figure 2 traveling between points Y and W).

**Lemma 4:** A ray can travel multiple times between two layers if the SSP has a local minimum between them.

**Proof:** A ray which is capable of traveling periodically between two layers has both maximum and minimum points of depth on its traveling path, as illustrated in Figure 2 between the points P and Q.

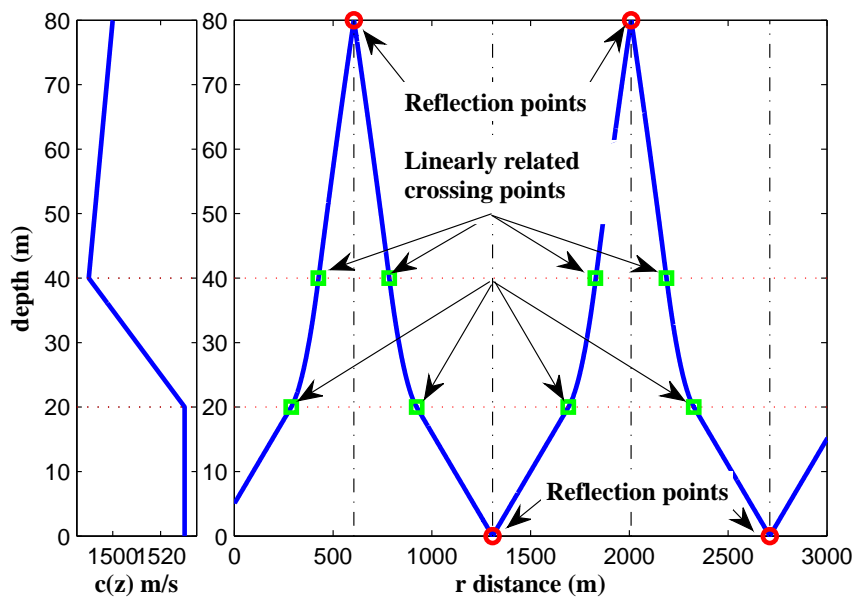
Therefore, based on Lemma 2, it must bend from the first layer to second one, and from the second layer to the first one. Since a ray has a tendency to bend toward the low speed region, periodic traveling between two layers happens if the lower speed region is located between the two layers. In addition, the assumption of an isogradient SSP for each layer forces the total SSP to have a local minimum value at the boundary of these two layers.

**Lemma 5:** Reflections from the seabed and sea surface can be formulated as a boundary equation. In a shallow underwater environment, it is very common that a traveling ray hits the sea surface or the seabed before reaching the end point. Based on the physical properties of the seabed and the sea surface, the reflection can be formulated as a boundary equation. For instance, if we consider a perfect reflection from the seabed or the sea surface, the boundary equations can be obtained as

$$\begin{aligned}\beta_{p+1} + \alpha_{p+1} &= -(\beta_p - \alpha_p) \\ \beta_{p+1} &= -\beta_p\end{aligned}\quad (20)$$

Under the assumption of a perfect reflection, the reflected parts in different layers have the same properties as the corresponding non-reflected parts but with an axial symmetry around the line parallel to the  $z$ -axis crossing the reflection point as illustrated in Figure 4. Due to this symmetry which is resulted by the cylindrical symmetry of the ray propagation, the  $r$ -coordinate of the two crossing points around a reflection point are linearly related to each other, and one can be formulated by the  $r$ -coordinate of the other. Therefore, this does not change the degree of the polynomial resulting from such kinds of ray patterns.

**Figure 4.** Linear dependency of the reflection and crossing points under the assumption of a perfect reflection.



Thanks to the piece-wise linear behavior of the SSP, we are now able to predict how a ray, which starts from a given point, can travel through different layers to arrive at a specific point. Having built a

ray pattern using the above lemmas we can then for every  $P$  part ray relate the ToF to the node's position using Equation (9) for each single-layer part and the following boundary equations

$$\frac{X_{p+1} + Y_{p+1}}{1 - K_{p+1}} = \frac{X_p - Y_p}{1 + K_p}, \text{ for } p = 1 \text{ to } P - 1 \quad (21)$$

The ToF of any possible  $P$  part ray can then be computed as

$$t = \sum_{p=1}^P \frac{-1}{a_p} \left[ \ln \frac{1 + \sin \theta_p^E}{\cos \theta_p^E} - \ln \frac{1 + \sin \theta_p^S}{\cos \theta_p^S} \right] \quad (22)$$

For instance, as illustrated in Figure 2, four kinds of rays can be predicted between the two points U and V which are described below.

- (i) A ray may travel directly from point U, crossing the boundary once, and not leaving the two adjacent layers. Of course, as formulated before, even with such a limitation, there are at most three paths that can be included in this category.
- (ii) A ray may travel directly from point U, crossing the boundary multiple times, not leaving the two adjacent layers. For three crossings of the ray with the boundary, a four-part ray will be obtained. Based on Equation (9c), the  $r$ -coordinate of the second crossing path is related to the previous one according to

$$r_2^E = r_1^E + 2 \frac{b_2 + a_2 z_{j-1}}{a_2} Y_2 \quad (23)$$

and similarly,

$$r_3^E = r_2^E + 2 \frac{b_3 + a_3 z_{j-1}}{a_3} Y_3 \quad (24)$$

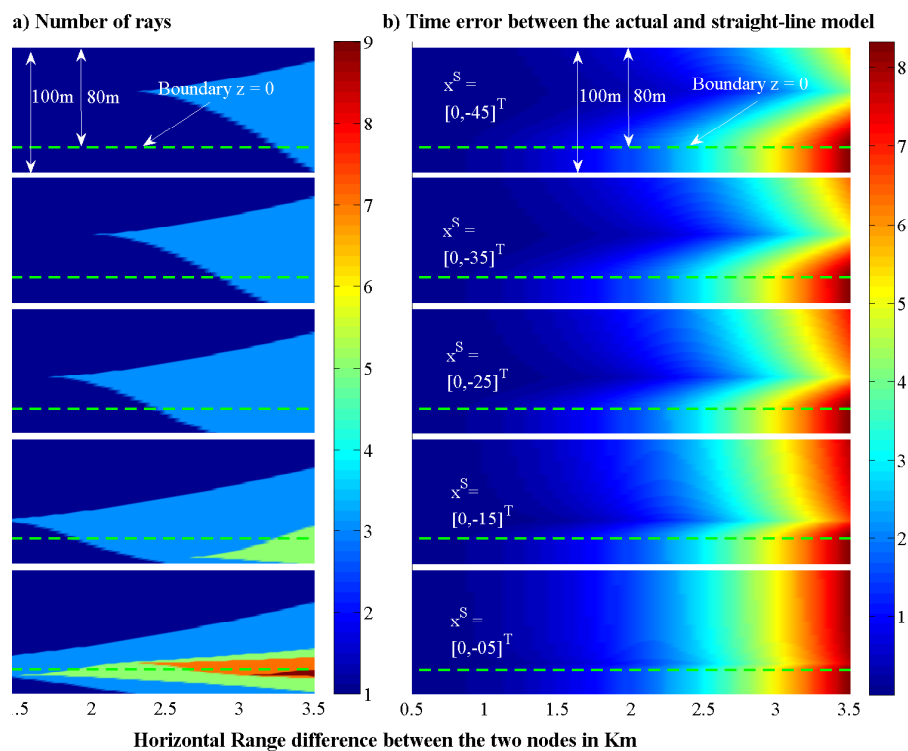
and based on the boundary condition we have  $Y_3 = -Y_2$  and  $X_3 = 0$ . From the above equations, it can be derived that the crossing points in this scenario are linearly dependent, and therefore the combination of all equations forms a fourth-order polynomial root finding problem owing to the single independent added boundary equation. For the other scenarios with more than three crossing points we have the same analysis, and the problem again reduces to a fourth-order polynomial root finding problem.

- (iii) It is also possible that a ray from U to V passes other layers. For instance, in the scenario depicted in Figure 2, the fourth layer has a positive steepness and a ray may bend over at that layer thereby entering a shallower layer. According to the introduced lemmas this is possible, and we should consider it in our analysis.
- (iv) Similar to the fourth layer, this phenomenon may also happen in the first layer which forms the fourth category of traveling paths.

In Figure 5(a), we show how many rays can travel between two points located in an unbounded two-layer underwater medium. Here, we imagine that the two layers have the same steepness but with different signs, e.g.,  $a_1 = -a_2 = 0.1$ , and consequently the SSP has a minimum value at the boundary of the two layers. The sound speed at the boundary ( $z = 0$ ) is assumed to be 1,480 m/s. To compute the number of possible rays between two points, we fix the position of  $\mathbf{x}^S$ , and change the position of  $\mathbf{x}^E$  to

cover a 100 m by 3.5 km area in the vertical plane as depicted in the figure. According to the discussed lemmas, the possible patterns that can propagate between these two points are 1.2, 1.2.1.2, 1.2.(1.2) . . . , or 1.1, 1.2.1, 1.(2.1) . . . , depending on where the two points are located. It is shown that as the pair-wise distance between the two nodes which are located close to the boundary of the two layers increases, the number of paths between them increases too. Around the region where SSP has a minimum value, a pattern with lower number of digits has a lower ToF, but a greater overshoot. Therefore, to compute the fastest ray in this region we can start searching with a simple pattern, and check Lemma 3. If Lemma 3 holds, then we can stop, otherwise we should continue with a more complicated pattern.

**Figure 5.** (a) Number of paths *versus* the location of the two nodes. (b) Range error due to the linear approximation of the ToF.



For the same scenario as described above, Figure 5(b) shows the error due to a linear approximation of the ToF. From the figure, it can be observed that as the distance between the two nodes gets larger, the linear approximation performs worse. In addition, another effect that influences the accuracy of the straight-line propagation is the angle of this line with the horizontal axis. The larger the angle of the straight line with the horizontal axis, the better the accuracy of the straight-line propagation model.

The above analysis indicates that if a UASN is forced to utilize a straight-line propagation model due to any reason (e.g., system complexity) to achieve more accurate localization results in a noise-free environment, it is suggested that sensors are deployed in such a way that the angles of the straight lines between the nodes become large. However, when noisy measurements exist, we should also consider their influence on the mapped horizontal distances.

#### 4. Pair-Wise Underwater Ranging

To our best knowledge, the algorithm in [11] is the only available mathematical approach for underwater range estimation. In order to better understand this algorithm we shortly review it here. The horizontal range between two nodes at different depths in an underwater medium can be obtained from Equations (6) and (7)

$$t = \int_{z^S}^{z^E} \frac{1}{c(z)} \frac{1}{\sqrt{1 - [k_0 c(z)]^2}} dz, \quad 0 < k_0 < \min_z \frac{1}{c(z)} \quad (25a)$$

$$r^E - r^S = \int_{z^S}^{z^E} \frac{k_0 c(z)}{\sqrt{1 - [k_0 c(z)]^2}} dz \quad (25b)$$

where  $k_0$  is a constant defined by Snell's law,  $t$  is the ToF between two nodes, and  $r^E - r^S$  represents the horizontal distance between them. The estimation of the horizontal distance has two phases; first, by measuring the depth and ToF information, the value of  $k_0$  can be computed numerically from Equation (25a), and second, by substitution of  $k_0$  into Equation (25b), and taking the integral, the value of  $r^E - r^S$  can be obtained. However, in an inhomogeneous medium, a ray trajectory is not always a monotonic function of the depth, and as a result, whenever a path between two nodes crosses a depth more than once, which is quite common, the above formulas are not valid anymore. In this case, either Equation (25a) has no answer for  $k_0$  in the specified range, or the obtained answer is not valid.

##### 4.1. Proposed Algorithm

Assume that, at a specific depth, the ToF of the fastest ray is a monotonic function of the horizontal range. In other words, a propagating wave at a specific depth reaches the destination with a smaller horizontal distance faster. Then, using the ToF and depth measurements, we can find the horizontal distance through a root finding algorithm such as Newton's method or bisection. Newton's method is very fast, but it requires the derivative of the ToF w.r.t. the horizontal distance which is hard to compute. The bisection method is robust, and it eventually finds the solution. However, it requires an upper and a lower bound on the horizontal distance. The lower bound can be set to zero, and the upper bound can be computed through multiplying the measured ToF by the maximum sound speed of the entire environment. In spite of the fact that other efficient numerical root-finding algorithms can also be used, we utilize the simple bisection algorithm for the results in the simulation section.

Algorithm 1 shows the steps of the proposed algorithm. In this algorithm,  $K$  and  $E$  are the user-defined limits on the stopping criteria that determine when the algorithm exits from the loop,  $r_{low}$  and  $r_{up}$  are the lower and the upper bound, respectively. The algorithm starts by initializing the upper and the lower bound on the range, and then it computes the fastest ToF for the midpoint of the bounds. In order to calculate the fastest ToF, given the depth of the two points, different ray-patterns that may host the fastest ray are formed, and all the rays between the points are found and their corresponding ToFs are computed, *i.e.*, in Algorithm 1,  $t^{[l]}$  represents the ToF of the  $l$ -th found ray between the points. Then, among all these ToFs, the smallest one is selected. Next, based on the computed ToF, the lower, the

upper, or both bounds are modified accordingly, and the procedure continues until one of the stopping criteria is met. The important factor that influences the complexity of the proposed algorithm is the number of ray patterns that may host the fastest ray. The ray patterns can be built very efficiently using the proposed Lemmas, still one can add more Lemmas (for a specific SPP) to reduce the number of ray patterns that may host the fastest ray between two nodes.

---

**Algorithm 1** Proposed Algorithm.
 

---

Compute horizontal distance upper and lower bounds,

$$r_{\text{low}} = 0,$$

$$r_{\text{up}} = \hat{t}c_{\text{max}}, \text{ where } c_{\text{max}} = \max_z c^{(j)}(z), j \in 1, \dots, N.$$

Initialize loop parameters,

$$e = r_{\text{up}} - r_{\text{low}},$$

$$k = 1.$$

**while**  $e \geq E$  and  $k \leq K$  **do**

  Compute the average value of the upper and the lower bound,

$$r = \frac{r_{\text{low}} + r_{\text{up}}}{2}.$$

  Find the smallest ToF for this horizontal distance

    - Form all possible ray patterns hosting the fastest ray (see lemmas).

    - Compute ToF for each possible ray  $t^{[l]}(r, \hat{z}^S, \hat{z}^E)$ , (see Equation (22)).

    - Select the ray with the smallest ToF.

$$t = \min_l t^{[l]}(r, \hat{z}^S, \hat{z}^E).$$

  Update the lower or the upper bound,

**if**  $t < \hat{t}$  **then**

$$r_{\text{low}} = r.$$

**else if**  $t > \hat{t}$  **then**

$$r_{\text{up}} = r.$$

**else**

$$r_{\text{low}} = r,$$

$$r_{\text{up}} = r.$$

**end if**

  Update loop parameters,

$$e = r_{\text{up}} - r_{\text{low}},$$

$$k = k + 1.$$

**end while**

  Compute the estimated horizontal distance between the nodes.  $\hat{r}^E - \hat{r}^S = \frac{r_{\text{low}} + r_{\text{up}}}{2}$

---

#### 4.2. Cramér–Rao Bound

The Cramér–Rao bound (CRB) expresses a lower bound on the variance of any unbiased estimator of a deterministic parameter. As mentioned before, since the depth information is known and the projection method can be used for localization, a given distance-based traditional localization algorithm works only with horizontal distances. Therefore, in this section we only derive the CRB for the horizontal distance estimation between two nodes. For the computation of the horizontal distance, three measurements are required: two depth measurements which are not directly related to the horizontal distance, and one ToF measurement. It is assumed that all the measurements are affected by Gaussian distributed noise as

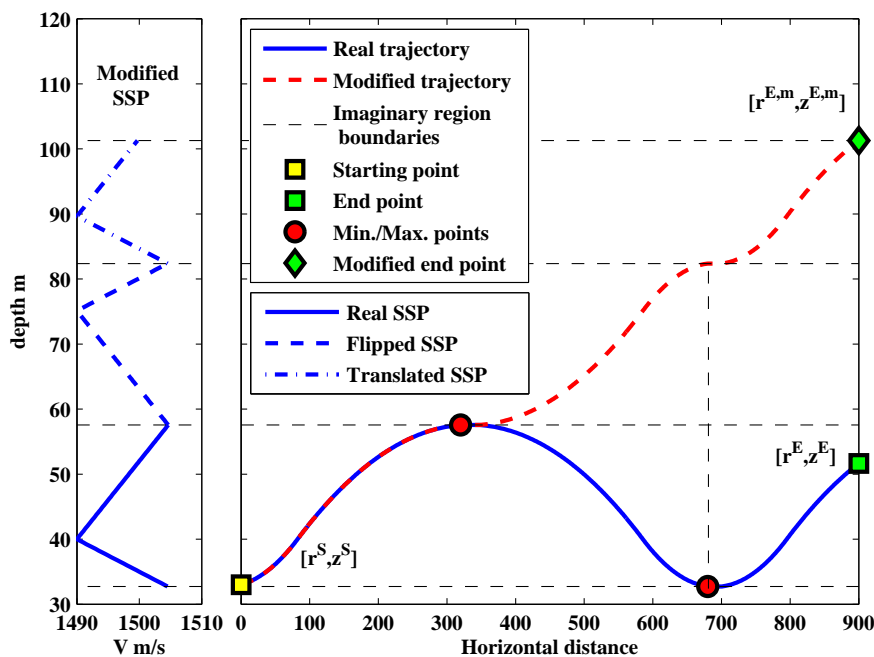
$$\begin{aligned}
 \hat{t} &= t + n_t \\
 \hat{z}^S &= z^S + n_z^S \\
 \hat{z}^E &= z^E + n_z^E
 \end{aligned}
 \tag{26}$$

where  $n_t, n_z^S$  and  $n_z^E$  are independent Gaussian distributed with variance  $\sigma_t^2, \sigma_z^S$  and  $\sigma_z^E$ , respectively. The Fisher information matrix for estimating the horizontal distance  $(r^E - r^S), z^S$ , and  $z^E$  can be obtained as

$$\mathbf{I}(r^E - r^S, z^S, z^E) = \frac{1}{\sigma_t^2} \begin{bmatrix} \frac{\partial t}{\partial(r^E - r^S)} \\ \frac{\partial t}{\partial z^S} \\ \frac{\partial t}{\partial z^E} \end{bmatrix} \begin{bmatrix} \frac{\partial t}{\partial(r^E - r^S)} & \frac{\partial t}{\partial z^S} & \frac{\partial t}{\partial z^E} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & \sigma_z^{-2} & 0 \\ 0 & 0 & \sigma_z^{-2} \end{bmatrix}
 \tag{27}$$

In order to compute the partial derivative  $\frac{\partial t}{\partial(r^E - r^S)}$ , we modify the environment in such a way that we can compute the horizontal distance as an integral w.r.t. depth. In order to achieve this, we have to convert the horizontal distance and the ToF to monotonic functions of the depth. Therefore, a ray can not have maximum or minimum points on its trajectory w.r.t. the depth. Let us illustrate the proposed idea with an example. Assume that a ray has a maximum point on its trajectory. The ray angle is zero at this maximum point, and after that it changes sign. But, this sign change does not affect Snell’s law, as it is related to the cosine of the ray angle. As a result, we can assume that the ray travels upward instead of downward as depicted in Figure 6, but in a new environment. In this new environment the SSP of each imaginary region must be changed accordingly. For instance, Figure 6 shows that the real SSP is flipped and translated in the first and second imaginary regions, respectively. In other words, the SSPs of the imaginary regions follow the behavior of the modified ray trajectory.

**Figure 6.** Changing the real ray trajectory into a trajectory which is a monotonic function of the depth.





Note that the above conversion can only be done after we compute the fastest ray, because only then we are able to locate the maximum and/or minimum points on the trajectory and build the new environment. Under this assumption, and using Equations (7b), (7b) and (6) we have

$$t = \int_{z^S}^{z^{E,m}} \frac{1}{c^m(z)} \frac{1}{\sqrt{1 - [k_0 c^m(z)]^2}} dz, \quad 0 < k_0 < \min_z \frac{1}{c^m(z)} \quad (28a)$$

$$r^E - r^S = \int_{z^S}^{z^{E,m}} \frac{k_0 c^m(z)}{\sqrt{1 - [k_0 c^m(z)]^2}} dz \quad (28b)$$

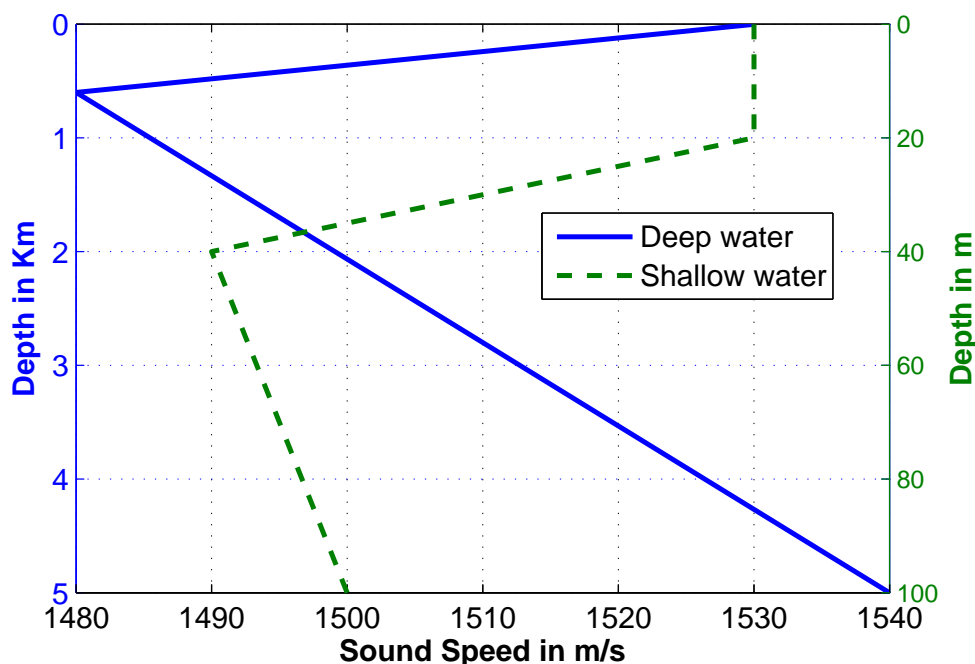
where m indicates that the variable is related to the modified environment. The above equations are similar to Equations (25a) and (25b), hence we can utilize the same approach used in [11] to compute the CRB, which results into

$$\text{var}(\hat{r}^E - \hat{r}^S) \geq \sigma_t^2 \frac{1}{k_0^2} + \sigma_z^2 \frac{1 - (k_0 c(z^E))^2}{(k_0 c(z^E))^2} + \sigma_z^2 \frac{1 - (k_0 c(z^S))^2}{(k_0 c(z^S))^2} \quad (29)$$

## 5. Numerical Results

In this section we study the performance of finding the fastest path, as well as the proposed ranging algorithm in a multi-layer underwater environment [15]. We consider two kinds of SSPs for our simulations as shown in Figure 7; the former is derived from the sound speed measurements in shallow water [16], and the latter is extracted from the sound speed of the Pacific Ocean and represents a deep water environment [13].

**Figure 7.** Sound speed profile for deep and shallow water.



### 5.1. Ray Propagation for Shallow Water

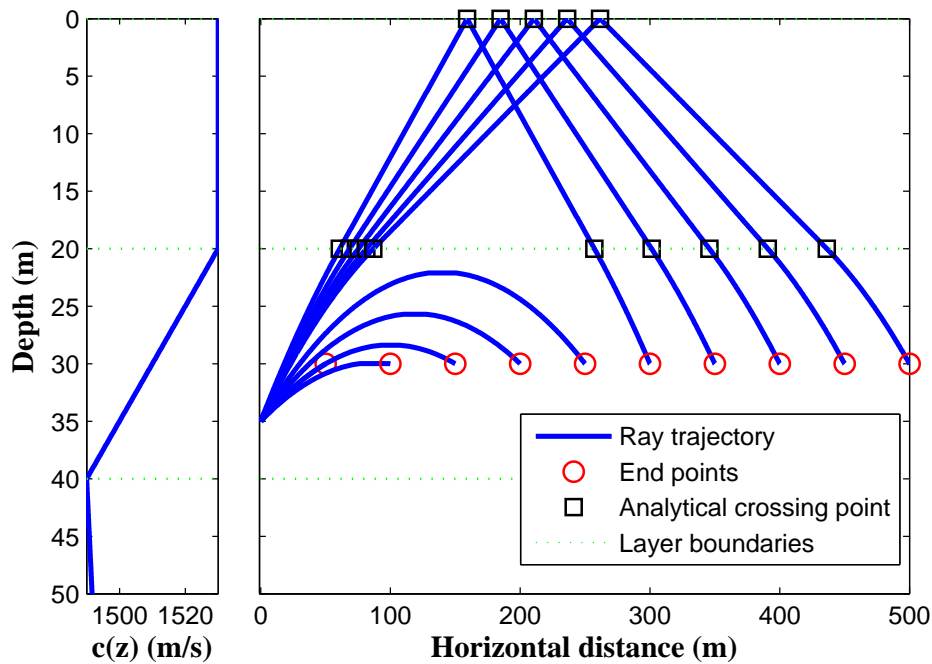
In this part of the report, based on the aforementioned lemmas, we analyze how a ray can propagate between two points inside the shallow water medium. Using the *ray pattern* concept, we are able to show how a ray can travel in a given medium, and which pattern may host the fastest ray. In Table 1, we show the family of patterns a ray may travel between two points through different layers. Since the depth of each node is known, we can select the proper patterns from the table, and form the corresponding polynomial formulas. By finding the roots of the polynomials, the ToF of each ray can be computed, and the fastest one will be recognized.

**Table 1.** All possible patterns that a fastest ray in a shallow underwater environment can follow.

	to layer 1	to layer 2	to layer 3
from layer 1	1	1.2	1.2.3
		1.1.2	1.1.2.3
		1.2.3.2	1.2.3.3
		1.2.3.3.2	1.2.3.2.3
from layer 2	2.1	2	2.3
	2.1.1	2.1.1.2	2.1.1.2.3
		2.3.2	2.3.3
		2.3.2.3.2	2.3.2.3
		⋮	⋮
from layer 3	3.2.1	3.2	3
	3.3.2.1	3.2.1.1.2	3.3
	3.3.2.1.1	3.2.3.2	3.2.3
		3.2.3.3.2	3.2.1.1.2.3
		3.2.3.2.3.2	3.2.3.2.3
	⋮	⋮	

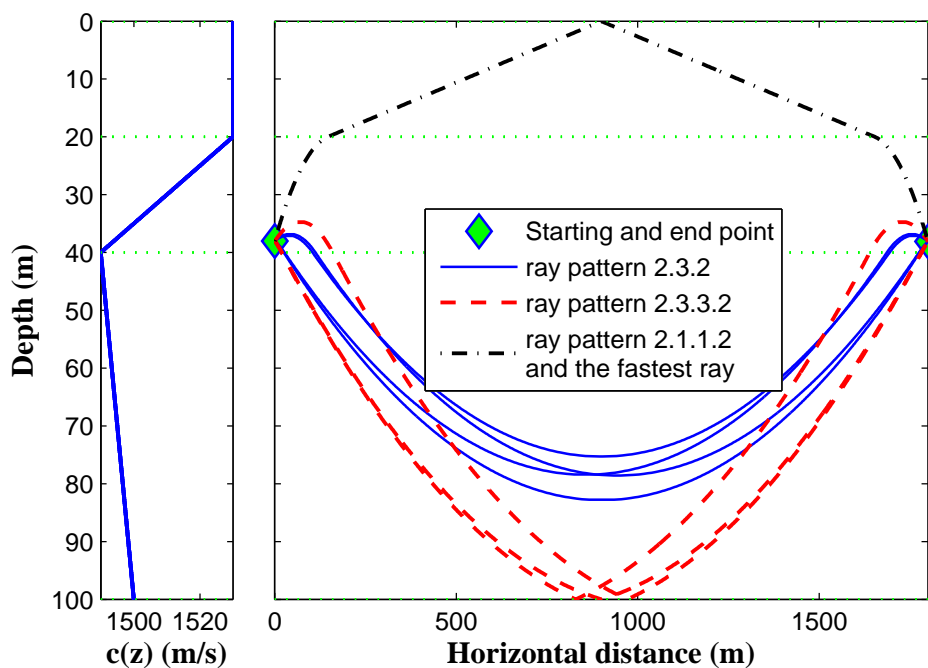
In Figure 8, we illustrate the ray propagation in a shallow underwater environment between two given points in the second layer. A ray can depart the first point and reach the second point in several ways; (a) it can directly propagate in the second layer without entering the other layers; (b) it can go to the upper layer, hit the surface and go back to the second layer; (c) it can go to the third layer and go back to the second layer; (d) it can go to the third layer, hit the bottom and go back to the second layer. Among all of these possibilities we choose the ray which has the lowest ToF. It is worth mentioning that the algorithm of [11] can not compute the correct horizontal range for any of the drawn blue-colored rays in Figure 8 except for the first one, since all other ray trajectories are not monotonic functions of depth.

**Figure 8.** Sample of ray propagation between two nodes.



In Figure 9, we show different possible rays that can travel between two points located in the second layer with a horizontal distance of 1,800 m. Based on the formulation, only three *ray patterns* can exist in this scenario, *i.e.*, 2.3.2, 2.3.3.2, and 2.1.1.2 (here we only consider one reflection from the surface, and only one reflection from the seabed in the existing ray patterns). Since the sound speed has higher values in the first and second layers, the fastest path belongs to the 2.1.1.2 pattern. It can be noted that if the horizontal distance between the two points increases, one *ray pattern* will be eliminated, namely 2.3.2.

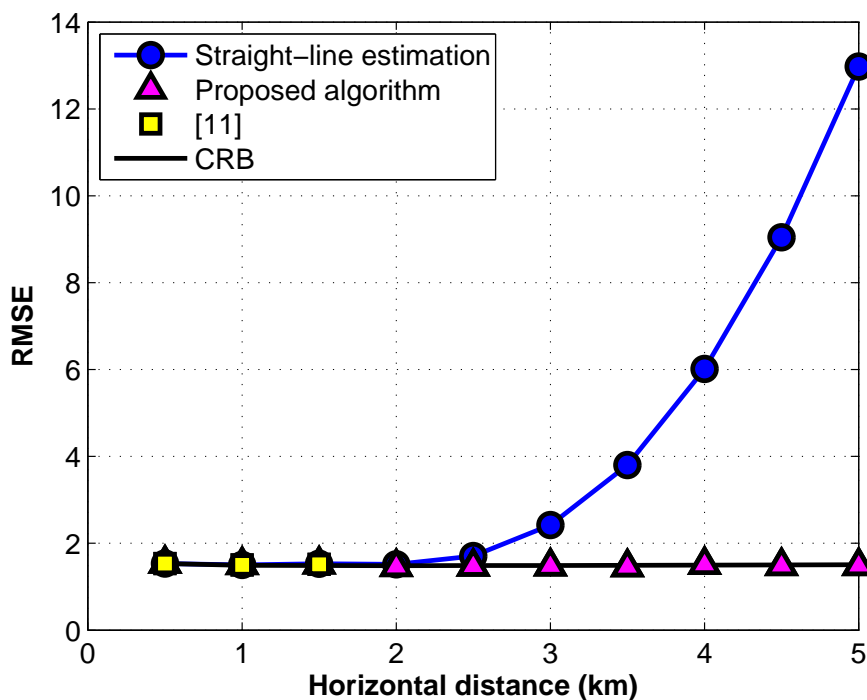
**Figure 9.** Different possible rays between two points in the second layer.



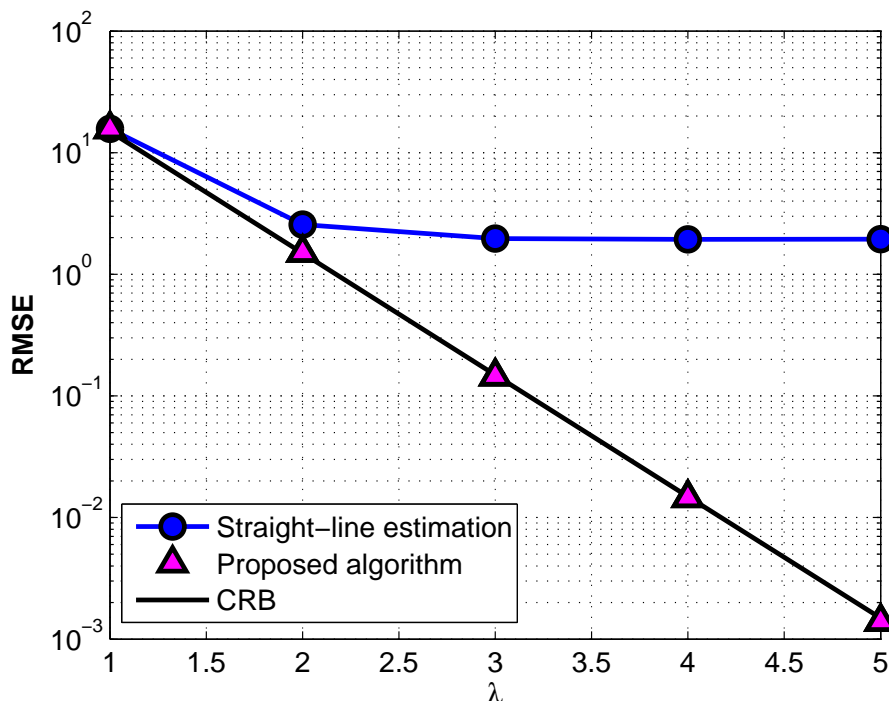
### 5.2. Ranging for Deep Water

In Figure 10, we compare the performance of the proposed range algorithm with the one introduced in [11], and with the algorithms which approximate the inhomogeneous underwater medium as a homogeneous one with a presumably constant sound speed, *i.e.*, we use a straight-line range computation based on the depth information. In this simulation, we consider Gaussian noise for the ToF and depth measurements with a standard deviation (std) of  $\sigma_t = 1$  ms and  $\sigma_z = 1$  m, respectively. In addition, we choose the deep water environment as a communication medium. The communication is between two points from different layers which are located at depth 550 m and 650 m, respectively. The root mean squared error (RMSE) for the horizontal distance estimation is computed by averaging over 1000 Monte Carlo simulations. As illustrated in this figure, the proposed algorithm performs well for all ranges while the algorithm of [11] has no definite solution from a given point as the horizontal range exceeds a given value. Furthermore, the straight-line algorithm degrades as the distance between the points increases.

**Figure 10.** Performance of the proposed algorithm for deep water.



In Figure 11, we investigate the effect of the measurement noise on the algorithms under consideration. Here the depth of the two nodes is as before, and their horizontal range is fixed at 3 km. The horizontal axis represents how noisy the measured data are. The depth and ToF measurement noise powers are exponentially related to the parameter  $\lambda$ , *i.e.*,  $\sigma_z = 10^3 \sigma_t = 10^{2-\lambda}$ . It can be seen that, the performance of the proposed ranging algorithm constantly improves and attains the CRB when we increase the measurement accuracy, while the straight-line estimation does not show any improvement after a given noise power.

**Figure 11.** Performance of the proposed algorithm for difference values of noise power.

## 6. Conclusions

We have analyzed the problem of localizing a target node in an underwater environment. The inhomogeneous underwater medium upsets the linear dependency of the pair-wise distances to the time of flight. We have shown that, if the depth information of the unlocalized node is available, then the problem of underwater localization can be converted to the traditional range-based one. Dividing the underwater medium into several isogradient sound speed profile layers, we have completely analyzed how a ray can travel between two given points through using different Lemmas. Further, we have proposed an iterative algorithm for the range estimation between two nodes, and we have demonstrated that the proposed algorithm attains the CRB and performs superb in comparison with other existing algorithms. In the future, we want to extend this work for more elaborate SSPs (not necessarily multiple isogradient), especially the ones with one local minimum, for ranging and channel modeling applications.

## Acknowledgements

The research leading to these results has received funding from the European Commission FP7-ICT Cognitive Systems, Interaction, and Robotics under the contract #270180 (NOPTILUS).

## References

1. Mao, G.; Fidan, B.; Anderson, B.D.O. Wireless sensor network localization techniques. *Comput. Netw. Int. J. Comput. Telecommun. Netw.* **2007**, *51*, 2529–2553.

2. Gerritsen, S. Waves in Inhomogeneous Media. Ph.D. Thesis, Delft University of Technology, Delft, The Netherlands, 2007.
3. Ameer, P.M.; Jacob, L. Localization using ray tracing for underwater acoustic sensor networks. *Commun. Lett. IEEE* **2010**, *14*, 930–932.
4. Ramezani, H.; Rad, H.J.; Leus, G. Localization and Tracking of a Mobile Target for an Isogradient Sound Speed Profile. In *Proceedings of the International Conference on Communications (ICC2012)*, Ottawa, Canada, 10–15 June 2012; accepted.
5. Coppens, A.B. Simple equations for the speed of sound in Neptunian waters. *J. Acoust. Soc. Am.* **1981**, *69*, 862–863.
6. Teymorian, A.Y.; Cheng, W.; Ma, L.; Cheng, X.; Lu, X. 3D underwater sensor network localization. *IEEE Trans. Mob. Comput.* **2009**, *8*, 1610–1621.
7. Isik, M.T.; Akan, O.B. A three dimensional localization algorithm for underwater acoustic sensor networks. *IEEE Trans. Wirel. Commun.* **2009**, *8*, 4457–4463.
8. Erol-Kantarci, M.; Mouftah, H.T.; Oktug, S. A survey of architectures and localization techniques for underwater acoustic sensor networks. *IEEE Commun. Surv. Tutor.* **2011**, *13*, 487–502 .
9. Tan, H.P.; Diamant, R.; Seah, W.K.; Waldmeyer, M. A survey of techniques and challenges in underwater localization. *Ocean Eng.* **2010**, submitted.
10. Casalino, G.; Caiti, A. RT<sup>2</sup>: A Real-Time Ray-Tracing Method for Acoustic Evaluations Among Cooperating AUVs. In *Proceedings of the OCEANS 2010 IEEE—Sydney*, Sydney, NSW, Australia, 24–27 May 2010; pp. 1–8.
11. Berger, C.R.; Zhou, S.; Willett, P.; Liu, L. Stratification effect compensation for improved underwater acoustic ranging. *IEEE Trans. Signal Process.* **2008**, *56*, 3779–3783.
12. Kussat, N.H.; Chadwell, C.D.; Zimmerman, R. Absolute positioning of an autonomous underwater vehicle using GPS and acoustic measurements. *IEEE J. Ocean. Eng.* **2005**, *30*, 153–164.
13. Porter, M.B. Acoustic models and sonar systems. *IEEE J. Ocean. Eng.* **1993**, *18*, 425–437.
14. Stojanovic, M.; Preisig, J. Underwater acoustic communication channels: Propagation models and statistical characterization. *IEEE Commun. Mag.* **2009**, *47*, 84–89.
15. Matlab codes for the simulations and algorithm are available online: <http://cas.et.tudelft.nl/~ramezani/PhDThesis.Codes.html> (accessed on 2 March 2012).
16. Gerstoft, P.; Hodgkiss, W.S. Effect of ocean sound speed uncertainty on matched-field geoacoustic inversion. *J. Acoust. Soc. Am.* **2008**, *123*, EL162–EL168.