

Proyecto de Investigación.  
20070053

## Móvil para Levantamiento Topográfico en Márgenes Acuáticos.

Araujo Díaz David (Profesor) y Antonio Hernández Yenifer Fabiola (Alumna PIFI).  
daraujo@ipn.mx y yeni\_whspr@yahoo.com.mx  
ESCOM-IPN.

### 1. Introducción.

El uso de ondas sonoras en el agua para la transmisión de información es de gran interés para el hombre. Una de las aplicaciones iniciales del **sonido subacuático** fue la instalación de timbres sumergidos en barcos-faro. El sonido subacuático de estos timbres podía detectarse a una distancia considerable por medio de un micrófono montado en el casco de un buque. Si se colocaban dos micrófonos en lados opuestos del casco y los sonidos recibidos eran transmitidos por separado a los oídos izquierdo y derecho, era posible determinar aproximadamente el rumbo del buque-faro. En **1912, Fessenden** desarrolló una fuente de sonido subacuática electrodinámica que permitía la comunicación entre barcos por medio de la **clave Morse**. La seguridad en la navegación oceánica fue mejorada con la introducción del **batímetro**, que determinaba la profundidad del agua midiendo el tiempo requerido para que pulsos cortos de sonido viajaran del transmisor al fondo del océano y regresaran.

El esfuerzo más grande en el sonido subacuático se ha asociado con la detección, seguimiento y clasificación de submarinos. Es costumbre aplicar el nombre **sonar (Sound Navigation and Ranging)** a esta fase de la acústica subacuática. Al enfrentar este problema ha sido necesario desarrollar medios para la conversión eficiente de potencia eléctrica en sonido subacuático y sistemas que son capaces de detectar señales débiles en la presencia de ruido. Ha sido igualmente importante el estudio de fenómenos fundamentales que afectan la transmisión del sonido en el océano, tales como la difusión, absorción, reflexión, refracción, dispersión, difracción y demás.

### 2. Objetivo del Proyecto.

El objetivo planteado para el Móvil para Levantamiento Topográfico en Márgenes Acuáticos fue el siguiente: Diseño y construcción de un dispositivo móvil para realizar el levantamiento topográfico en márgenes acuáticos, así como el desarrollo del software necesario para su interpretación y graficación en dos y tres dimensiones.

### 3. Velocidad del sonido en el agua de mar.

La velocidad del sonido en el **agua pura** es función de la **temperatura** y la **presión**. Un factor adicional en el **agua de mar** es la **salinidad**. **Lovett** ha analizado críticamente una gran colección de mediciones de laboratorio de la velocidad del sonido en el agua con diferentes salinidades y a varias temperaturas y presiones, y postuló una elaborada ecuación empírica. Una aproximación razonablemente exacta a esta ecuación es:

$$c(L, S, t) = 1449.05 + 45.7 t - 5.21 t^2 + 0.23 t^3 + (1.333 - 0.126 t + 0.009 t^2) (S - 35) + ?(L) \quad \dots \text{Ec. (1)}$$

$$\text{Donde: } ?(L) = 16.3 L + 0.18 L^2$$

Esto da **c(L, S, t)** para una latitud de **45°**. Para otras latitudes, la cantidad **L** debe reemplazarse con **L(1 - 0.0026 cos Ø)**, donde **Ø** es la latitud en grados. En la **Ec. (1)**, **S** es la salinidad en partes por millar (pp millar), **L** es la profundidad en kilómetros, y **t = T / 10** donde **T** está en grados Celsius. La **Ec. (1)** con corrección por latitud tiene una desviación normal de **0.06 m/seg** con respecto a la ecuación de **Lovett** cuando se aplica hasta una profundidad de **4 km** en aguas oceánicas, excluyendo el mar Negro, mar Rojo, el Mediterráneo y el Golfo Pérsico. Si se requiere una **?(L)** más exacta, pero más complicada, la aproximación es:

$$? (L) = (16.23 + 0.253 t) L + (0.213 - 0.1 t) L^2 + [0.016 + 0.0002(S - 35)](S - 35)t L \quad \dots \text{Ec. (2)}$$

Es válida prácticamente sobre todas las aguas oceánicas hasta una profundidad de **4 km** con una desviación normal de **0.02 m/seg**. Se debe aplicar la misma corrección por latitud. Nótese que la velocidad del sonido en la superficie del **agua de mar** con una salinidad de **35 pp millar** es de **1,449 m/seg** a **0°C** en contraste con **1,403 m/seg** para **agua fresca** bajo las mismas condiciones de presión y temperatura.

En muchos cálculos es frecuentemente adecuado usar una velocidad nominal de **1,500 m/seg** que es típica de las medidas en aguas superficiales que cubren las placas continentales en latitudes medias. A esta velocidad nominal está asociada la impedancia característica  $r_o c = 1.54 \times 10^6 \text{ Pa.s/m}$ . Será práctica usual, en lo que sigue, usar los valores anteriores para obtener las relaciones numéricas entre la presión acústica, velocidad de partícula y la intensidad en el agua de mar. Sin embargo, en cálculos que requieren diferencias en la velocidad del sonido, se debe usar la **Ec. (1)** o una ecuación más exacta.

#### 4. Pérdida sonora por transmisión.

La pérdida por transmisión se define como:

$$TL = 20 \log [P(1) / (r)] \quad \dots \text{Ec. (3)}$$

Donde **P(r)** y **P(1)** son las **amplitudes de la presión acústica** medidas a las **distancias horizontales r** y **1 m** de la fuente sonora.

Por ejemplo, la amplitud de una **onda esférica amortiguada** es:

$$P(r) = (A / r) e^{-a(r-1)} \quad \dots \text{Ec. (4)}$$

Donde **a** es el **coeficiente de absorción en nepers por metro**. Tomando el logaritmo en ambos lados:

$$20 \log P(r) = 20 \log P(1) - 20 \log r - a(r - 1) \quad \dots \text{Ec. (5)}$$

Donde **a = 8.7a** es el coeficiente de absorción en **decibeles por metro**. Ya que para todas las frecuencias de interés **a < 1 dB/m**, la pérdida por transmisión para difusión esférica con absorción es:

$$TL = 20 \log r + ar \quad \dots \text{Ec. (6)}$$

Si el sonido está atrapado entre dos superficies perfectamente reflejantes, la pérdida por transmisión es:

$$TL = 10 \log r + ar \quad \dots \text{Ec. (7)}$$

La **Ec. (7)** es la **difusión cilíndrica con absorción**.

En general, es conveniente separar la pérdida por transmisión en dos partes:

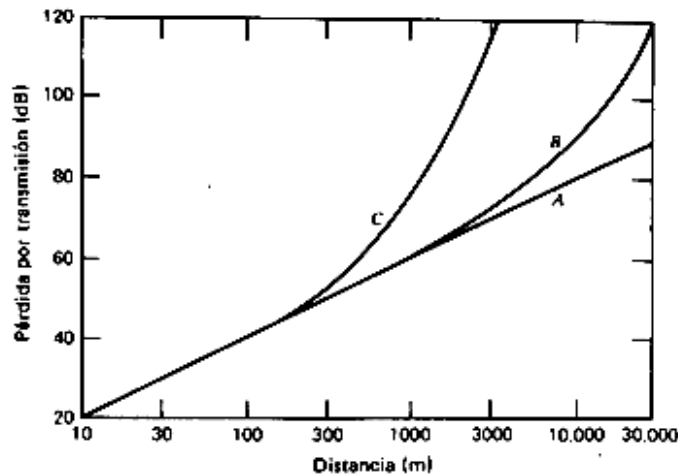
$$TL = TL(\text{geométrica}) + TL(\text{pérdidas}) \quad \dots \text{Ec. (8)}$$

Donde **TL(geométrica)** representa la pérdida por consideraciones geométricas y **TL(pérdidas)** es la pérdida debida a la absorción, dispersión y otros efectos no-geométricos. Con **TL(geométrica) = 20 log r** ó **10 log r** y **TL(pérdida) = ar**.

El **coeficiente de absorción a** para ondas sonoras en agua de mar, a una presión de **1 atmósfera** y una **temperatura de 5°C**, es **a = 0.00006 dB/m** a **1 kHz**, **0.0008 dB/m** a **10kHz**, y **0.013 dB/m** a **50 kHz**. Una aproximación para agua de mar a **5°C** y **una atmósfera** (profundidad cero) que es adecuada es:

$$a / F^2 = [8 \times 10^{-5} / (0.7 + F^2)] + [0.04 / (6000 + F^2)] + 4 \times 10^{-7} \quad \dots \text{Ec. (9)}$$

Donde  $F$  es la frecuencia en kilohertz y  $a$  está dada en decibeles por metro. En la **Figura 1** se encuentran graficadas curvas que muestran la **pérdida por transmisión** para **propagación esférica** con absorción en función de  $r$  para cada una de las frecuencias anteriores. A bajas frecuencias y distancias cortas, la pérdida por transmisión se debe fundamentalmente a la propagación esférica. Conforme la frecuencia y el alcance aumentan, las **curvas B y C** muestran que las pérdidas por absorción cobran más importancia. Es evidente que para lograr la transmisión a grandes distancias es necesario emplear bajas frecuencias.



**Figura 1:** Dependencia de la pérdida por transmisión  $TL$  para difusión esférica con absorción con la distancia y la frecuencia. **A** a 1 kHz, **B** a 10 kHz y **C** a 50 kHz.

Cuando se hacen medidas de pérdidas por transmisión en el océano, se observa que a menudo se desvían de las predichas por la **Ec. (6)**. Los factores que intervienen en esto incluyen:

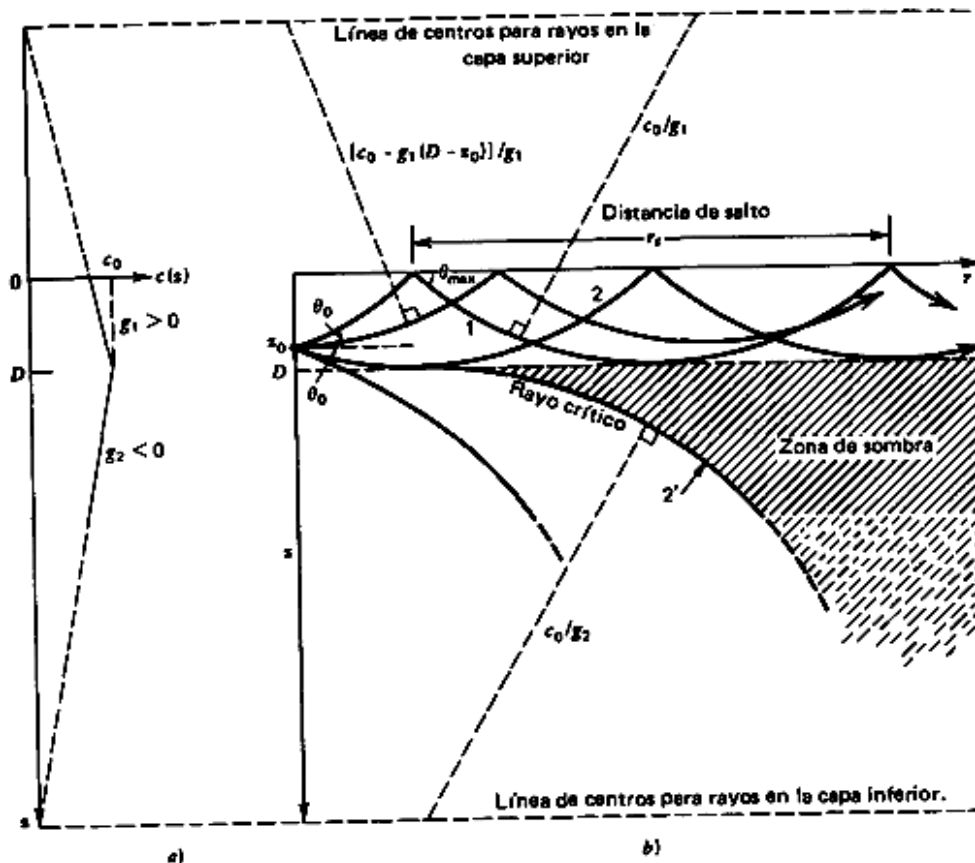
- 1) Efectos geométricos debidos a la divergencia o convergencia causada por la refracción, o interferencia constructiva o destructiva asociada con la propagación en múltiples trayectorias incluyendo reflexiones de la superficie y del fondo del océano.
- 2) Atenuación aumentada por la difracción y dispersión causada por inhomogeneidades del agua.

Aunque en condiciones idealizadas es posible derivar ecuaciones y calcular valores precisos para la pérdida por transmisión asociada con cada uno de estos factores, los océanos son tan complicados que por lo común se utilizan ya sea modelos analíticos simples o bien modelos complicados por computadora para calcular la pérdida por transmisión en cualquier situación real.

Aun cuando muchos factores limitan la capacidad de transmitir el sonido a través del agua, debe notarse que el sonido es inmersamente superior a las ondas electromagnéticas para transmitir energía a través del agua de mar. Por ejemplo, las ondas de radio de frecuencia más baja en uso comercial, **30 kHz**, se atenúan **1 dB** en **0.3 m**, y las frecuencias más altas se atenúan aún más rápidamente. De igual manera, la difusión y dispersión de un rayo de luz que pasa a través del agua es tan grande que el medio es, para todo propósito práctico, opaco a distancias arriba de **200 m**, y los rayos más penetrantes se reducen en **1 dB** por cada **1.5 cm** de trayectoria. En comparación con otros medios disponibles, el uso de ondas sonoras para transmitir energía a través del agua de mar es superior, y sólo se degrada cuando se contrasta con la transmisión, mucho más eficiente, de ondas de radio y luz a través del aire.

## 5. Capa mezclada.

Como se mencionó anteriormente, la acción de las olas superficiales puede causar una mezcla del agua en la capa superficial, formando lo que se conoce como la **capa mezclada**. El gradiente de velocidad positivo en esta capa mezclada atrapa al sonido cerca de la superficie. Una vez formada, la capa mezclada tiende a permanecer hasta que la energía procedente del Sol empieza a calentar la parte superior, haciendo decrecer el gradiente. Este efecto de calentamiento culmina en un gradiente negativo que lleva a una refracción hacia abajo y la pérdida del sonido de la capa. Como esto ocurre por lo general en la tarde, el efecto se conoce como **efecto de atardecer**, durante la noche, el enfriamiento de la superficie y la mezcla debida a las olas permiten que se reestablezca la capa isotérmica. Es raro que se establezca un gradiente positivo mayor que aproximadamente **0.016/seg** debido a que esto requiere que la temperatura aumente con la profundidad, una condición dinámicamente inestable ya que, para salinidad constante, la densidad decrecería con la profundidad. Cuando la capa mezclada está presente, el perfil sonoro cerca de la superficie puede modelarse con dos gradientes lineales, como se muestra en la **Figura 2a**, donde **D** es la profundidad de la capa. La **Figura 2b** muestra rayos representativos de una fuente en la capa a una profundidad  $z_0$ . Un rayo que viaja hacia arriba se refleja de la interfase agua-aire con un ángulo de reflexión igual al ángulo de incidencia, y un rayo que intercepta la frontera inferior de la capa continúa en una trayectoria determinada por el gradiente abajo de la capa.



**Figura 2:** Transmisión sonora en una capa mezclada.

Nótese que la trayectoria se continúa suavemente sin ningún cambio en  $q$ . Todos los rayos que dejan la fuente con ángulos de elevación o depresión entre los de los rayos 1 y 2 se confinarán a la capa mezclada. Los rayos 1 y 2 tienen radio de curvatura y son tangentes al fondo de la capa. El rayo 2' se llama **rayo crítico** ya que demarca la frontera interior de la zona de sombra, dentro de la cual no se encuentra ningún rayo. Aunque este modelo simple indica que no hay señal en la zona de sombra, esto es incorrecto. La dispersión de las

burbujas y la superficie oceánica áspera, la presencia de ondas internas que hacen que  $D$  fluctúe con la distancia horizontal, y la difracción del sonido de la periferia de la zona de sombra hacia su interior, contribuyen a dar una zonificación débil y fluctuante. A frecuencias altas en la zona de muchos **kilohertz**, los niveles de las señales están típicamente por lo menos **40 dB** abajo de los de las orillas de la zona. Para frecuencias bajas en **kilohertz**, la pérdida de señal en la zona de sombra es menos severa, y para frecuencias suficientemente bajas la zona de sombra puede dejar de existir debido a la fuerte difracción y la invalidez de la teoría de rayos.

Todos los rayos cuyos ángulos de elevación o depresión exceden los de los **rayos 1 y 2** penetran a mayores profundidades y se salen de la capa. Finalmente, los rayos entre los rayos límites (1 y 2) que inicialmente se propagan esféricamente serán atrapados en la capa y se propagarán entonces cilíndricamente. La distancia a la cual ocurre el cambio de propagación esférica a propagación cilíndrica se llama la **distancia de transición**  $r_t$ . Esto puede estimarse para  $q_0$  pequeño, requiriendo que cuando  $r = r_t$  el ancho vertical del haz subtendido por  $2q_2$  iguale la profundidad de la capa  $D$ ,  $r_t = D / (2 q_0)$ .

Otro parámetro importante es la **distancia de salto**  $r_s$  de la **Figura 2b**,  $r_s = 2q_{\text{máx}}$  donde  $R = c_0 / g_1$  es el radio del rayo que raza el fondo de la capa y  $R > D$ . Para ángulos pequeños, la **ley de Snell** da:

$$1 / [c(D)] = 1 / [c(0)] (1 - 0.5 q^2_{\text{máx}}) \quad \dots\text{Ec. (10)}$$

Y puede aproximarse la velocidad del sonido con:

$$c(z) = c(0) (1 - z / R) \quad \dots\text{Ec. (11)}$$

Combinando las anteriores ecuaciones se obtiene:

$$r_s = 2 \sqrt{2RD} \quad \dots\text{Ec. (12)}$$

$$r_t = (1 / 8) r_s \sqrt{D / (D - z_0)} \quad \dots\text{Ec. (13)}$$

El valor nominal de  $R$  en la capa mezclada es  $R = 1,500 / 0.016 = 9.4 \times 10^4 \text{ m}$ .

Con la ayuda de estas cantidades, se puede obtener un **modelo de pérdida por transmisión simple**. Para  $r < r_t$  la propagación geométrica es **esférica**, y para  $r > r_t$  la propagación es **cilíndrica**. Formas adecuadas que se acoplan a la distancia de transición son:

$$TL_{\text{geométrica}} = \begin{cases} 20 \log r & r < r_t \\ 10 \log r + 10 \log r_t & r > r_t \end{cases} \quad \dots\text{Ec. (14)}$$

Hay varias contribuciones a **TL(pérdidas)**. La absorción del sonido por el agua de mar la representa  $ar$ . También hay pérdidas del ducto superficial por dispersión, ondas internas e irregularidades en la velocidad del perfil sonoro. Estas contribuciones pueden parametrizarse haciendo que dependan de la distancia de salto y una **pérdida por rebote**  $b$ . Al combinar se obtiene **TL(pérdidas) =  $ar + br / r_s$** . En consecuencia, la pérdida por transmisión es:

$$TL = \begin{cases} 20 \log r + (a + b / r_s)r & r < r_t \\ 10 \log r + 10 \log r_t + (a + b / r_s)r & r > r_t \end{cases} \quad \dots\text{Ec. (15)}$$

Aunque todos los rayos atrapados en el ducto pueden subir a la superficie para reflejarse otra vez hacia abajo, no todos los rayos pueden llegar al fondo del canal. Por ejemplo, un rayo que deja a la fuente horizontalmente nunca puede adquirir una profundidad mayor que la de la fuente. Si un receptor está a una profundidad  $Z_r$  mayor que la de la fuente, sólo puede detectar aquellos rayos que alcanzan profundidades iguales o mayores que el mismo. Esto significa que la pérdida por transmisión entre una fuente y un receptor más profundo debe ser mayor que la que se obtiene de la **Ec. (6)**. Recuerde que la reciprocidad acústica establece que el intercambio del receptor y la fuente no puede alterar la pérdida por transmisión entre dos

puntos. Si se intercambian la fuente y el receptor, la fuente estará en la profundidad  $Z_r$  original y el alcance de transición está ahora determinado por la **Ec. (14)** por  $Z_r$ , en lugar de  $Z_0$ . Este valor excede al calculado antes, por lo que la pérdida por transmisión es mayor. Al calcular  $r_t$  se debe usar la mayor de las profundidades del receptor y la fuente.

Dado que la pérdida por transmisión depende de la profundidad de la fuente (o receptor, según el que esté a más profundidad), la pérdida por transmisión en un ducto dado se minimiza haciendo que el receptor y la fuente estén poco profundos. Sin embargo, ninguno de ellos puede estar a unas cuantas longitudes de onda de las superficies, ya que la interferencia entre cualquiera de ellos y su imagen pueden volverse importantes.

Aunque la pérdida por rebote está sujeta a considerables variaciones dependiendo de las propiedades de la capa mezclada, algunos valores de  $b$  aproximados pero representativos pueden obtenerse a partir de una ecuación desarrollada empíricamente por **Schulkin**:

$$b = (SS)F^{1/2} \quad \dots\text{Ec. (16)}$$

Donde **SS** es el estado del mar, una clasificación de la aspereza de la superficie (**Tabla 1**), y **F** es la frecuencia en kilohertz. La gama de validez de esta ecuación es para  $3 < b < 14$  dB/rebote en el intervalo de frecuencias de **2 kHz a 25 kHz**.

**Notas: 1)** Velocidad del viento para las alturas medias indicadas. **2)** Mar totalmente encrespado.

Descripción	Altura de las olas comúnmente aceptada	Estado del Mar	Altura media de las olas	Viento 12 -horas (nudos)	Viento (nudos)	Tiempo requerido (H)	Agarre requerido (millas náuticas)
Mar como espejo	-	0	-	-	-	-	-
Rizos con forma de escamas, pero sin crestas espumosas	-	1/2	-	2	2	2	-
Olas pequeñas, cortas pero pronunciadas, crestas tienen apariencia cristalina, pero no rompen.	0-1	1	0.7	5	5	7	40
Olas grandes, crestas empiezan a romper, espuma de apariencia cristalina, cabrillas esparcidas.	1-2	2	2	9	9	11	100
Olas, pequeñas, más largas; cabrillas frecuentes.	2-4	3	3.5	14	13	14	150
Olas moderadas, que toman una forma más pronunciada (posibilidad de salpicadura).	4-8	4	6	19	17	18	200
Se empiezan a formar grandes olas, las crestas blancas espumosas son más extensas (probablemente algunas salpicaduras).	8-13	5	9.5	24	21	23	300
El mar se levanta y la espuma blanca de las olas que rompen es llevada por el viento.	13-20	6	13.5	30	25	28	400
Olas moderadamente altas de mayor longitud; las orillas de las crestas se rompen en forma de rocío; la espuma es llevada por el viento en chorros bien marcados.	20-30	7	18	36	24	32	500

**Tabla 1:** Parámetros usados en la caracterización de propiedades superficiales.

Debido a que el confinamiento del sonido en la capa mezclada establece una guía de ondas cilíndrica, a frecuencias más bajas se debe usar la teoría de modos normales en vez de la teoría de rayos. Existen modos

normales cada uno con su propia frecuencia de corte. Esto significa que para frecuencias suficientemente bajas, muchos de los modos normales son desvanecientes y la capacidad de la capa mezclada para transportar energía se ve reducida. Por consiguiente, el confinamiento del sonido en la capa mezclada no ocurre para sonidos de longitud de onda suficientemente larga. Esta frecuencia de corte depende del gradiente abajo de la capa mezclada, pero una aproximación a la frecuencia abajo de la cual el sonido no quedará bien atrapado está dada por la **Ec. (17)**.

$$f @ 2 \times 10^5 / D^{3/2} \quad \dots \text{Ec. (17)}$$

Donde la profundidad **D** está en **metros** y la frecuencia **f** en **hertz**. Para frecuencias cercanas a este valor o menores, el modelo de pérdida por transmisión de la capa mezclada es cada vez más sospechoso, y los modelos de modos normales dan predicciones más exactas.

Los rayos cuyos ángulos de elevación o de depresión exceden  $q_0$  en la fuente se pierden por la capa mezclada y se refractan hacia abajo. Estos rayos se reflejan del fondo o, debido a la capa isotérmica profunda, se refractan hacia arriba hasta llegar a la superficie. A menudo los rayos refractados convergen cuando llegan a la superficie, lo cual lleva a un aumento en los niveles de la señal en la superficie e inmediatamente abajo de ella. A esto se le denomina **zona de convergencia**, y la distancia a la cual ocurre depende de los detalles del perfil de velocidad sonora y varía mucho de un océano a otro. La zona de convergencia usualmente abarca entre **15 y 70 km** y su ancho (la distancia sobre la cual hay mejoramiento significativo de la señal) usualmente es del **10 por ciento** del alcance de la zona de convergencia. La reflexión de rayos del fondo de la superficie constituye también una trayectoria útil en el océano. Estos rayos ayudan a aumentar el intervalo entre los límites exteriores de propagación en la capa mezclada y cualquier zona de convergencia. Sin embargo, las incertidumbres de las propiedades reflectivas del fondo del océano limitan la utilidad de las trayectorias de propagación de rebote en el fondo.

## 6. Ecuaciones del Sonar.

En todas las aplicaciones de sonido subacuático (ya sea que se quiera detectar y localizar objetos bajo el agua, encontrar peces, etc.) la operación crítica es la detección de la señal acústica deseada en presencia de ruido. Si el nivel de la señal es el nivel de eco **EL** y el nivel de ruido el nivel de ruido detectado **DNL**, entonces la ecuación del sonar es:

$$EL \geq DNL + DT \quad \dots \text{Ec. (18)}$$

El umbral de detección **DT**, es el valor por el cual el nivel de eco debe exceder el nivel de ruido detectado para dar una probabilidad del **50** por ciento de detección para una probabilidad de falsa alarma dada. En el procesamiento de señales para sonido subacuático, se usa el índice de detección **d** para especificar los umbrales de detección en lugar del índice de detectabilidad **d'**. Estos índices están relacionados de una manera simple.

$$d = (d')^2 \quad \dots \text{Ec. (19)}$$

### a) Sonar pasivo.

Un sistema que escucha el ruido producido por el "blanco" se llama un sistema de sonar pasivo. (En este caso, el uso del término "nivel de eco" no es apropiado, pero su uso es convencional). El sonido radiado por el blanco a un nivel de fuente **SL** experimenta una pérdida por transmisión **TL** en su camino al receptor. El nivel de eco es entonces:

$$EL = SL - TL \quad \dots \text{Ec. (20)}$$

Con un detector altamente directivo, un sistema pasivo puede determinar la dirección de donde llega la señal. Si el blanco puede detectarse simultáneamente en dos o más receptores separados por una distancia conocida, se puede localizar al blanco por triangulación.

El ruido de una gran variedad de fuentes compite con la señal recibida. Los océanos están llenos de fuentes de ruido como las olas que rompen, chasquidos de canoas y barcos que se combinan para producir un

ruido ambiental de banda ancha. Adem3s, se produce ruido propio por la maquinaria en la plataforma de recepci3n y por el movimiento del agua a su alrededor. El nivel combinado de estas fuentes da el nivel de ruido **NL**. Si el receptor es direccional, el nivel de ruido detectado es:

$$DNL = NL - DI \quad \dots Ec. (21)$$

Donde el 3ndice de directividad **DI** describe la habilidad del receptor para discriminar entre el ruido proveniente de otras direcciones distintas a la de la fuente. Combinando las ecuaciones **Ecs. 19, 20** y **21** se obtiene la ecuaci3n del sonar pasivo:

$$SI - TL \cong NL - DI + DT \quad \dots Ec. (22)$$

#### b) Sonar activo.

Para un sistema activo, la se3al es un pulso de energ3a ac3stica que se origina en el transmisor con un nivel de fuente **SL**. Esta se3al viaja despu3s al blanco, acumulando una p3rdida por transmisi3n en un solo sentido **TL**. En el blanco, una fracci3n de la se3al incidente, caracterizada por una intensidad del blanco **TS**, se refleja hacia la fuente y, sufriendo una segunda p3rdida por transmisi3n **TL'**, llega al receptor. Para el caso monoest3tico, la fuente y el receptor est3n en la misma posici3n, por lo cual **TL = TL'** y el nivel de eco es:

$$EL = SL - 2TL + TS \quad \dots Ec. (23)$$

Al determinar el tiempo **t** entre la emisi3n de un pulso y el regreso del eco, puede encontrarse la distancia **r** al blanco de **r = ct / 2**. Si el receptor es altamente direccional, tambi3n se puede determinar la localizaci3n del blanco.

El nivel de ruido detectado para un sistema activo puede ser dominado por el ruido ambiente o el ruido propio. Entonces:

$$SL - 2TL + TS \cong NL - DI + DT \quad \dots Ec. (24)$$

Para el sonar activo hay una fuente adicional de enmascaramiento que no est3 presente en el sonar pasivo: la **reverberaci3n**. La reverberaci3n surge de la dispersi3n de la se3al emitida por blancos indeseables, tales como peces, burbujas as3 como la superficie del mar y del fondo. Para este caso, el nivel de ruido detectado es el nivel de reverberaci3n **RL**:

$$DNL = RL \quad \dots Ec. (25)$$

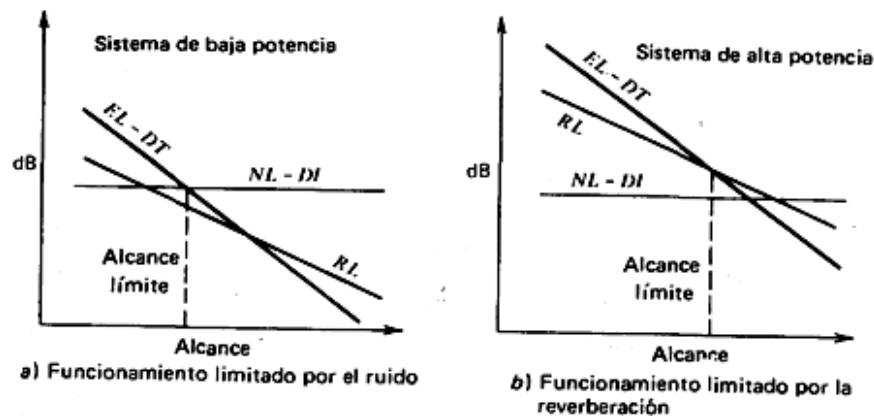
Al combinar **Ecs. 27, 31** y **44** se obtiene la ecuaci3n para (monoest3tico) el sonar activo limitado por la reverberaci3n:

$$SL - 2TL + TS \cong RL + DT \quad \dots Ec. (26)$$

El que el sistema de sonar activo est3 dominado por el ruido o por la reverberaci3n depende de la potencia ac3stica, el alcance y la velocidad del blanco.

Las dos posibles situaciones se muestran en la **Figura 3**. Generalmente, los sistemas de baja potencia est3n limitados por el ruido, ya que el alcance m3ximo de detecci3n se logra cuando el nivel del eco cae abajo del nivel al cual puede extraerse del ruido (**Figura 3a**). El aumento de la potencia ac3stica de un sistema aumenta tanto el nivel del eco como el nivel de reverberaci3n a una distancia dada, pero generalmente la reverberaci3n disminuye al aumentar la distancia con menos rapidez que el nivel del eco (**Figura 3b**). Si a medida que la distancia aumenta el eco disminuye hasta que est3 enterrado en la reverberaci3n, se dice que el sistema est3 limitado por la reverberaci3n.





**Figura 3:** Funcionamiento de un sistema de sonar limitado por el ruido y la reverberación.

Una estrategia empleada frecuentemente para reducir el efecto de la reverberación es usar un filtro de incisión en el receptor para eliminar la energía en una banda de frecuencia estrecha que comprende la reverberación. Si el blanco se mueve, la frecuencia del eco será diferente del de la reverberación y se puede detectar más fácilmente el blanco. Sin embargo, el filtro anterior también eliminará al eco si el blanco es estacionario con respecto al agua.

## 7. Consideraciones de Ruido y Amplitud de Banda.

De las ecuaciones del sonar es claro que el funcionamiento de éste puede mejorarse si se reduce el nivel del ruido detectado. Esto se puede lograr si se conoce el espectro de frecuencias del ruido ambiente y del blanco para seleccionar la amplitud de banda del sistema de recepción.

### a) Ruido ambiente.

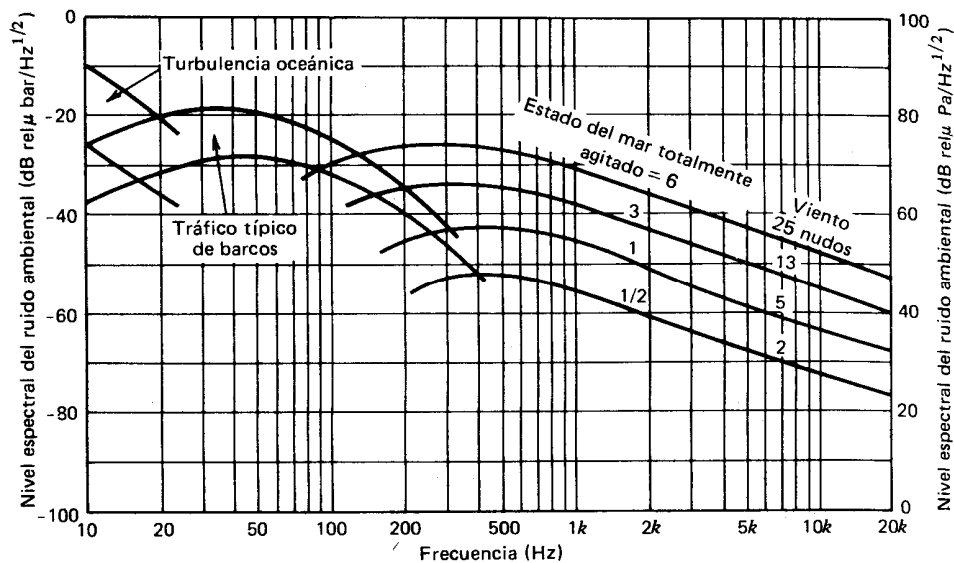
La forma normal del nivel espectral de ruido **NSL** ambiente en el océano abierto se representa en la **Figura 4**. Entre aproximadamente **500 Hz** y **20 kHz**, la agitación de la superficie local del mar es la fuente más fuerte de ruido ambiental puede caracterizarse especificando la velocidad del viento local. Las relaciones entre el estado del mar, la altura promedio de las olas y las velocidades del viento representativas se dan en la **Tabla 1**. En este intervalo de frecuencias el nivel espectral del ruido cae a aproximadamente **17 dB/década**. A frecuencias más bajas, la mayor contribución al ruido ambiental es causado por los barcos distantes y fuentes biológicas. Los límites indicados en la figura pueden excederse considerablemente si el tránsito de barcos es grande. Abajo de **20 Hz** predominan la turbulencia oceánica y el ruido sísmico. Arriba de **50 kHz** la agitación térmica de las moléculas del agua se vuelve una fuente importante de ruido, y el nivel espectral del ruido aumenta a razón de **6 dB/octava**. En agua poco profunda, los niveles de ruido pueden ser considerablemente mayores debido al mayor tránsito de barcos, ruido biológico ambiental, ruidos provenientes de las orillas, plataformas de perforación, etc.

El nivel del ruido espectral en la figura se midió con receptores omnidireccionales. Si se usa un receptor direccional, se verá que el ruido de la superficie del mar llega predominantemente de la dirección vertical, mientras que el ruido de los barcos proviene de la dirección horizontal. Por consiguiente, el nivel de ruido detectado por un receptor direccional depende de su orientación.

El nivel de ruido detectado para el ruido ambiental y si el ancho de banda es lo suficientemente pequeño, entonces

$$DNL = NSL + 10 \log w - DI$$

...Ec. (27)



**Figura 4:** Ruido ambiente de aguas profundas.

#### b) Ruido propio.

El ruido propio es generado por la plataforma de recepción e interfiere con la señal recibida deseada. El ruido propio puede llegar al receptor por transmisión a través de la estructura mecánica y por transmisión (a través del agua), ya sea directamente de la fuente o por reflexión de la superficie del mar. El ruido propio usualmente tiende a aumentar con la velocidad de la plataforma. A bajas frecuencias y bajas velocidades, domina el ruido de las máquinas, mientras que a altas frecuencias el ruido de la propela y de flujo adquieren importancia. Conforme se aumenta la velocidad, estas últimas fuentes son más importantes a todas las frecuencias. A velocidades muy bajas, usualmente el ruido propio es menos importante que el ruido ambiental o la reverberación. A velocidades mayores el ruido propio puede ser el factor limitante.

El ruido propio se introduce en las ecuaciones del sonar como un nivel espectral de ruido isotrópico equivalente, que expresa el nivel de enmascaramiento del ruido propio en el ancho de banda del receptor en términos del nivel de una cantidad equivalente de ruido ambiental.

#### 1) Sonar pasivo.

Considérense dos buques que viajan en direcciones diferentes con velocidades diferentes. Si el **buque 1** origina una señal de frecuencia  $f_1$ , entonces por el argumento anterior un observador estacionario en el agua a un ángulo  $q$  con respecto al movimiento del buque fuente percibirá una señal de frecuencia:

$$f_w = f_1 \left( 1 + \frac{V \cos q}{c} \right)$$

El **buque 2** en movimiento recibirá esta señal y observará una frecuencia:

$$f_2 = f_w \left( 1 + \frac{U \cos f}{c} \right)$$

Eliminando a  $f_w$  de estas dos ecuaciones y la suposición  $U \ll c$  y  $V \ll c$  da:

$$\Delta f = f_2 - f_1 = \frac{R}{c} f_1$$

Donde la rapidez de alcance:

$$R = \frac{dR}{dt} = V \cos \mathbf{q} + U \cos \mathbf{f}$$

Es la velocidad con que los dos buques varían su distancia. Si la fuente y el receptor se acercan uno al otro,  $R$  es positiva y la señal recibida aumenta de frecuencia (**Doppler** hacia arriba). Si la fuente y el receptor se separan uno del otro, la señal recibida baja de frecuencia (**Doppler** hacia abajo).

## II) Sonar activo.

Supóngase que la señal  $f_1$  del **buque 1** es un pulso de sonar activo. Debido a que el **buque 2** se mueve con respecto al agua, el eco del **buque 2** tendrá una frecuencia  $f'_w$  en el agua.

$$f'_w = f_2 \left( 1 + \frac{U \cos \mathbf{f}}{c} \right)$$

Y el **buque 1** recibirá un eco de frecuencia:

$$f'_1 = \left( 1 + 2 \frac{R}{c} \right) f_1$$

Por otro lado, la reverberación proviene de dispersores en reposo en el agua, y la frecuencia de reverberación  $f_r$  como la observa el **buque 1** es:

$$f_r = f_1 \left( 1 + 2 \frac{V \cos \mathbf{q}}{c} \right)$$

El **buque 1** puede comparar la frecuencia  $f'_1$  del eco recibido ya sea con  $f_1$  (su frecuencia sonar) o con  $f_r$  (la frecuencia de reverberación). Los corrimientos **Doppler** correspondientes son:

$$\Delta f_1 = f'_1 - f_1 = 2 \frac{R}{c} f_1$$

$$\Delta f_r = f'_1 - f_r = 2 \frac{U \cos \mathbf{f}}{c} f_1$$

### c) Consideraciones de amplitud de banda.

La presencia de un corrimiento **Doppler**  $Df_1$  del eco con respecto a la fuente impone un límite inferior en la amplitud de banda del receptor ya que el filtro de paso de banda no debe ignorar los ecos de blancos que se mueven rápidamente. Si se conoce la máxima rapidez, la amplitud de banda debe ser el doble del corrimiento **Doppler** asociado (ya que el blanco puede acercarse o alejarse). Por consiguiente, un receptor de un sonar activo debe tener una amplitud de banda total de:

$$w = 2.67 \times 10^3 R f \quad (\text{activo})$$

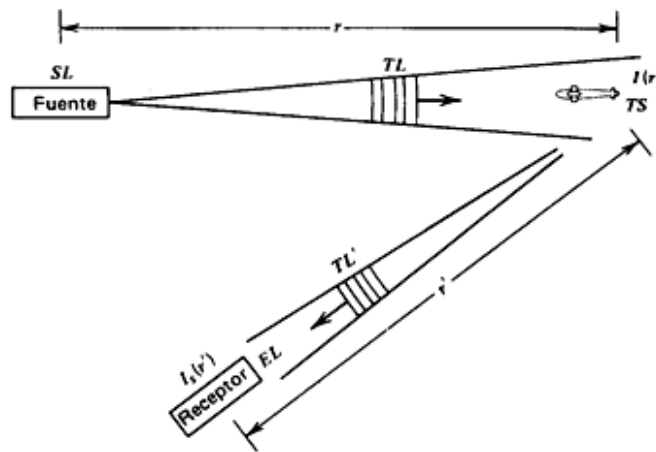
...Ec. (28)

Donde  $w$  y  $f$  están en **hertz** y en **metros por segundo**. Para un sistema pasivo, diseñado para detectar una frecuencia específica, debe de tener una amplitud total igual a la anterior:

$$w = 1.3 \times 10^{-3} R f \quad (\text{pasivo}) \quad \dots \text{Ec. (29)}$$

**Sonar activo.**

**a) Fuerza de un blanco.** Una fuente acústica manda un pulso en el océano que intercepta un blanco y lo ilumina con una intensidad  $I(r)$ . El blanco dispersa el sonido en todas direcciones, y algo se refleja en la dirección del receptor. Para el receptor, el blanco ha generado (por reflexión) una señal acústica que se propaga del blanco al receptor. Este proceso se muestra en la **Figura 5**.



**Figura 5:** Diagrama usado al derivar la expresión para la fuerza del blanco.

Si la señal reflejada  $I_s(r')$  se extrapola del campo lejano a **1 m** del centro acústico del blanco ( $r' = 1$ ), la razón  $I_s(r' = 1) / I(r)$  mide la capacidad del blanco para reflejar sonido incidente hacia el receptor. Es común escribir esta razón en la forma:

$$\frac{I_s(r' = 1)}{I(r)} = \frac{s}{4p} \quad \dots \text{Ec. (30)}$$

Donde  $r$  es la sección transversal acústica del blanco.

El nivel del eco en el receptor es:

$$EL = 10 \log [I_s(r' = 1) / I_{ref}] - TL' \quad \dots \text{Ec. (31)}$$

$$EL = 10 \log \left[ \frac{I(r) \cdot s}{I_{ref} \cdot 4p} \right] - TL' \quad \dots \text{Ec. (32)}$$

Sabiendo que  $TL = 10 \log [I(1) / I(r)]$  y que  $SL = 10 \log [I(1) / I_{ref}]$  se ve que  $r$  y  $TS:1$

$$TS = 10 \log \frac{r}{4p} \quad \dots \text{Ec. (33)}$$

El primer término a la derecha se puede interpretarse como un nivel de fuente aparente:

$$sL = 10 \log \frac{I(r)}{I_{ref}} + TS \quad \dots \text{Ec. (34)}$$

Y el nivel del eco enviado al receptor es:

$$EL = sL - TL' \quad \dots \text{Ec. (35)}$$

La fuerza de blanco de un objeto reflector está determinada fundamentalmente por su tamaño, forma y construcción, así como por la frecuencia del sonido incidente. Por ejemplo, se puede demostrar que la fuerza de blanco de una esfera perfectamente reflectora de radio  $a$  metros, que rerradía la energía sonora interceptada uniformemente en todas direcciones (a  $1$ ), está dada por

$$TS = 20 \log(a / 2) \quad \dots \text{Ec. (36)}$$

Esta ecuación indica que una esfera de **2 m** de radio tendrá una fuerza de blanco de **0 dB**. Un blanco de **0 dB** corresponde a uno que rerradía el sonido con un nivel de fuente efectivo que es igual al nivel de presión del sonido incidente. Para esferas mayores, el nivel de fuente rerradiado es mayor que el nivel del sonido incidente. Se puede esperar que la fuerza de blanco de un objeto de forma irregular, tal como un submarino, dependa de su orientación con respecto al sonido incidente. Por ejemplo, las fuerzas de blanco de los submarinos tipo flota de la Segunda Guerra Mundial están a un mínimo alrededor de **10 dB** para sonidos incidentes en la proa y popa, y aumenta hasta alrededor de **25 dB** por la manga.

Un pulso incidente cuya longitud en el agua sea corta comparada con la longitud del blanco retornará como un eco mucho mayor conforme se refleja primero del borde de entrada del blanco y continúa reflejándose hasta que pasa el borde de salida. Este ensanchamiento del pulso reduce la intensidad del eco debido a que la energía acústica en el eco se esparce sobre un intervalo de tiempo mayor.

Poco se sabe acerca de la dependencia de la frecuencia en la fuerza de blanco para la mayoría de los blancos. Sin embargo, las altas frecuencias son más favorables para clasificar un blanco porque las ondas cortas permiten que algo de la estructura del blanco se observe en el eco recibido, en tanto que para ondas largas se pierde mucho detalle en el eco. Los pulsos muy cortos y de alta frecuencia mostrarán reflexiones de varias partes del blanco como ecos discretos o encimados, en tanto que un pulso largo se parecerá más a la fuerza de blanco medida con excitación continua.

La estela generada por las propelas del blanco puede contribuir a su detectabilidad al crear una región turbulenta llena de burbujas que genera considerables señales dispersas. La importancia de la estela es una función fuerte de la velocidad y profundidad del blanco, y tiende a ser muy débil para un submarino lento y profundo. Una fuerza de estela típica está entre **0** y **-30 dB** por cada metro de longitud iluminado.

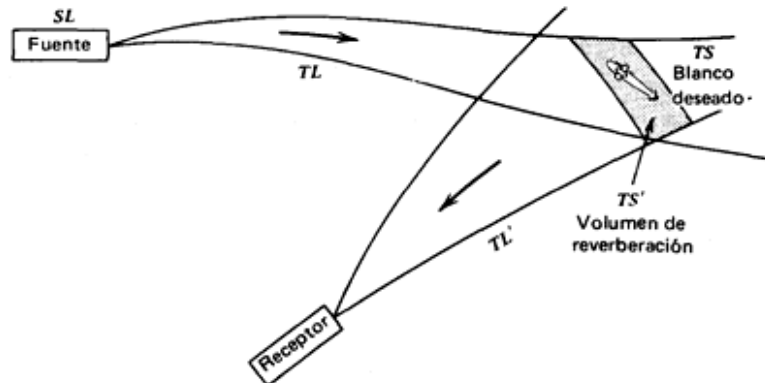
**b) Reverberación.** Cuando una fuente activa sonoriza alguna porción del océano, puede haber dispersión por las burbujas, partículas, peces, la superficie o el fondo del mar y cualesquiera otras inhomogeneidades presentes. Estas constituyen fuentes de señales no deseadas, es decir reverberación, que pueden interferir con el eco del blanco de interés.

Un paso esencial para obtener el **nivel de reverberación RL** es calcular el **volumen V** (o **superficie A**), a la distancia del blanco, del cual el sonido dispersado puede llegar al receptor durante el mismo tiempo que el eco del blanco deseado. Obviamente, esto dependerá de la longitud del pulso, las directividades de la fuente y el receptor, y de la geometría. Dado este volumen (o superficie), se podrá calcular directamente el nivel de reverberación. Por el momento, se supondrá que se conoce **V** (o **A**) y se obtendrán las fórmulas para **RL**.

La **Figura 6** muestra el blanco y el volumen reverberante circundante que puede dispersar el sonido que interfiere con el eco en el receptor. La intensidad **I(r)** de la señal que ilumina esta región está relacionada con el nivel de fuente **SL** del transmisor por:

$$10\log \frac{I(r)}{I_{ref}} = SL - TL \quad \dots Ec. (37)$$

Donde **TL** es la pérdida por transmisión de la fuente al blanco.



**Figura 6:** Diagrama usado para derivar el nivel de reverberación para dispersores de volumen.

Cada dispersor dentro del volumen reverberante tiene un nivel de fuente aparente dado por:

$$sL_i = 10\log \left[ \frac{I(r) s_i}{I_{ref} 4p} \right] \quad \dots Ec. (38)$$

Donde  $r_i$  es la sección transversal acústica del dispersor  $i$ . Suponiendo que los dispersores individuales tienen fase aleatoria, la intensidad total dispersada es la suma de las intensidades individuales

$$sL = 10\log \left[ \frac{I(r)}{I_{ref}} \sum_v \frac{s_i}{4p} \right] \quad \dots Ec. (39)$$

Donde la sumatoria cubre a todos los dispersores en  $V$ .

La fuerza de dispersión  $S_v$  para un volumen unitario se define como:

$$S_v = 10\log \left[ \frac{1}{V} \sum_v \frac{s_i}{4p} \right] = 10\log \frac{S_v}{4p} \quad \dots Ec. (40)$$

Donde  $S_v$  es la sección transversal total por unidad de volumen:

$$s_v = \frac{1}{V} \sum_v s_i \quad \dots Ec. (41)$$

Por consiguiente:

$$sL = 10\log \frac{I(r)}{I_{ref}} + S_v + 10\log v \quad \dots Ec. (42)$$

Análogamente, si la reverberación viene de una superficie, se puede definir una fuerza de dispersión  $S_A$  por unidad de área superficial. El nivel de sonido dispersado por la superficie es entonces:

$$sL = 10 \log \frac{I(r)}{I_{ref}} + S_A + 10 \log A \quad \dots \text{Ec. (43)}$$

Pueden interpretarse como fuerzas de dispersión para la región reverberante:

$$TS_R = \begin{cases} 10 \log V + S_v \\ 10 \log A + S_A \end{cases} \quad \dots \text{Ec. (44)}$$

El nivel de reverberación en el receptor es:

$$RL = sL - TL' \quad \dots \text{Ec. (45)}$$

Si la fuente y el receptor están juntos entonces  $TL = TL'$ :

$$RL = SL - 2 TL + TS_R \quad \dots \text{Ec. (46)}$$

Otra forma es:

$$TS \cong TSR + DT \quad \dots \text{Ec. (47)}$$

Esto muestra claramente que la interferencia proveniente de la reverberación con la señal deseada no es más que la competencia de blancos no deseados (dispersores). Una característica sobresaliente de estas expresiones es que son independientes de la fuerza de la fuente. Por consiguiente, una vez que el  $SL$  es lo suficientemente grande para que la reverberación se vuelva más importante que el ruido, no hay ninguna ventaja al aumentar más el  $SL$ . Se deben dirigir esfuerzos para reducir  $DI$  o  $DT$  para obtener una mejoría en el alcance de la detección.

## 8. Canal de agua de poca profundidad de isovelocidad.

En esta sección se exploran algunas técnicas más complicadas que el trazo de rayos. En muchos casos, cuando la velocidad del sonido es una constante, es posible aplicar el poderoso, pero engorroso, método de imágenes. La interferencia superficial que se vio anteriormente, constituye un ejemplo en donde se sustituye el efecto de la superficie con una imagen  $180^\circ$  fuera de fase con la fuente. El método es válido para todas las frecuencias; recuérdese que la teoría de rayos surge de la ecuación Eikonal, una aproximación a la ecuación de ondas válida únicamente a altas frecuencias. La solución por este método usualmente se expresa como una suma infinita o finita de las contribuciones de todas las imágenes. Un análisis adecuado se lleva a cabo a menudo con la ayuda de una computadora digital.

En esta sección se formularán las soluciones a tres problemas de propagación por el método de imágenes, pero después se harán varias aproximaciones para obtener expresiones simples en forma cerrada para la pérdida por transmisión. Esto revelará la física subyacente y al mismo tiempo dará una base para comparar con los resultados de la teoría de rayos y con el análisis de modos normales.

Como consecuencia de la multiplicidad de reflexiones que tienen lugar tanto en la superficie como en el fondo, las predicciones de la pérdida por transmisión son más complicadas en aguas poco profundas que para el caso de aguas profundas. Según la teoría de imágenes, la superficie y el fondo de la capa de isovelocidad pueden tratarse como interfases a través de las cuales el sonido se propaga con la amplitud de presión reducida por el coeficiente de reflexión para esa superficie y el ángulo de incidencia. Esto da un espacio de multicapas (**Figura 7**), en donde las diferentes imágenes de la fuente contribuyen al punto del campo a lo largo de trayectorias de líneas rectas. Se han etiquetado las imágenes con el número de veces que la señal de la imagen al punto del campo se refleja del fondo. Se pueden obtener modelos de pérdida por transmisión simples si se hacen algunas aproximaciones. Se supone que la superficie del océano es un reflector perfecto. Si se supone que las inhomogeneidades en el agua y la aspereza de la superficie y el fondo son suficientes como para que las fases de las diferentes imágenes sean aleatorias, ocurre una combinación incoherente de las

diversas contribuciones y pueden ignorar las fases relativas. Además, se puede suponer que cada conjunto de cuatro imágenes cuyas contribuciones hacen  $i$  intersecciones con el fondo yacen a una distancia efectiva  $r_i = \sqrt{r^2 + (2iH)^2}$  del punto del campo. Esto permite que las imágenes se sumen en grupos de cuatro.

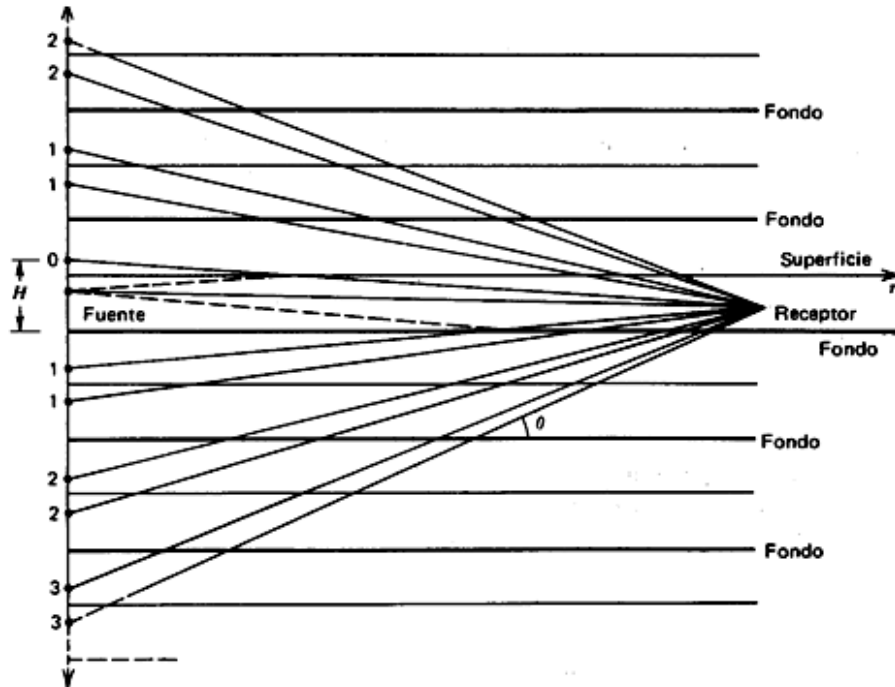


Figura 7: Fuente e imágenes para un canal de isovelocidad de poco fondo.

$$\frac{I(r)}{I(1)} = \frac{2}{r^2} + 4 \sum_{i=1}^{\infty} \frac{R^{2i}(q_i)}{r^2 + (2iH)^2} = \frac{2}{r^2} (1 + S) \quad \dots \text{Ec. (48)}$$

Donde se define a  $S$  por:

$$S = \sum_{i=1}^{\infty} \frac{R^{2i}(q_i)}{1 + i^2 (2H/r)^2} \quad \dots \text{Ec. (49)}$$

Cada ángulo de incidencia  $q$  se evalúa con

$$\cos q_i = 1 / \sqrt{1 + i^2 (2H/r)^2} \quad \dots \text{Ec. (50)}$$

Y  $R(q_i)$  es el coeficiente de reflexión para el fondo cuando el ángulo de incidencia es  $q_i$ .

Si la distancia de la fuente al receptor es mucho mayor que la profundidad del canal,  $H/r \ll 1$ , la sumatoria puede reemplazarse por la integral:

$$S = \int_1^{\infty} \cos^2 q R^{2u}(q) du \quad \dots \text{Ec. (51)}$$

Donde la variable de integración  $u$  ha reemplazado a  $i$ , y  $q$  es una función de  $u$ . De la Ec. (87) se ve que  $u = 1/2 (r/H) \tan q$ , y al cambiar la variable de integración  $u$  a  $q$  se obtiene:



$$S = \frac{r}{2H} \int_{\tan^{-1}(2H/r)}^{p/2} R^{(r/H) \tan q} (q) dq \quad \dots \text{Ec. (52)}$$

a) **Fondo perfectamente rígido.** Si el coeficiente de reflexión de presión en el fondo es la unidad para toda  $q$ , el integrando la **Ec. (89)** es la unidad para toda  $q$  y  $r \gg H$ .

$$S = \frac{r}{2H} \int_{r/2H/r}^{p/2} dq = \frac{p}{4} \frac{r}{H} - 1 \quad \dots \text{Ec. (53)}$$

Al sustituir en la **Ec. (85)** se obtiene:

$$\text{TL(geométrica)} = 10 \log r + 10 \log (H / p) \quad \dots \text{Ec. (54)}$$

La divergencia es cilíndrica y hay una contribución a **TL** parecida a la  $r_t$  para el ducto superficial.

b) **Fondo lento.** Si el fondo tiene una velocidad del sonido menor que la correspondiente al agua, el coeficiente de reflexión es pequeño a ángulos de incidencia grandes y aumenta hasta la unidad a incidencia rasante. En consecuencia, la mayor contribución a **S** provendrá del límite inferior de la integración. Para ángulos de incidencia pequeños se puede escribir el coeficiente de reflexión como  $R(q) \sim \exp(-g q)$  donde  $g$  es un parámetro que se determina a partir de las características del fondo. Puesto que la mayor parte del valor de **S** proviene de  $q \sim 2H / r$ , se reemplazará  $\tan q$  por  $q$  y se hará que el límite superior de la integración se vuelva infinitamente grande. Entonces:

$$S = \frac{r}{2H} \int_{r/2H/r}^{p/2} e^{-(r/H)gq^2} dq \quad \dots \text{Ec. (55)}$$

Un cambio de variable de  $q$  a  $x = q(g r / h)^{1/2}$  da:

$$S = \frac{1}{2} \sqrt{\frac{r}{gH}} \int_{r/2Hg/r}^{\infty} e^{-x^2} dx \quad \dots \text{Ec. (56)}$$

Para  $r$  grandes, el límite inferior tiende a **0** y  $\int_0^{\infty} \exp(-x^2) dx = \sqrt{p} / 2$ . Al sustituir, se obtiene:

$$\frac{I(r)}{I(1)} = \frac{2}{r^2} \left( 1 + \frac{1}{2} \sqrt{\frac{pr}{gH}} \right) \quad \dots \text{Ec. (57)}$$

Y en el límite de grandes distancias esto lleva a

$$\text{TL(geométrica)} = 15 \log r + 5 \log (g H / p) \quad \dots \text{Ec. (58)}$$

Por consiguiente, el efecto de un fondo lento es dar una propagación entre la esférica y la cilíndrica.

c) **Fondo rápido.** En este caso el coeficiente de reflexión es idénticamente igual a la unidad para todos los ángulos rasantes menores que el ángulo crítico  $q_c$ . Todas las trayectorias que se reflejan de fondo con ángulos que exceden a  $q_c$  sufren pérdidas y se atenúan más rápidamente que en la forma cilíndrica. Las trayectorias más rasantes que  $q_c$  serán atrapadas y se esparcirán cilíndricamente a distancias grandes. Debido a estas consideraciones, la **Ec. (53)** puede evaluarse integrando del límite inferior a  $q_c$  y haciendo  $R(q) = 1$  dentro de esa distancia. Con la restricción  $r \gg 2H / q_c$  para que el límite inferior pueda ser reemplazado con cero, **S** se vuelve:

$$S = \frac{r}{2H} \int_0^{q_c} dq = \frac{r}{2H} q_c \quad \dots \text{Ec. (59)}$$

Y por consiguiente:

$$TL(\text{geométrica}) = 10 \log r + 10 \log (H / (2 q_c)) \quad \dots \text{Ec. (60)}$$

Nótese que  $q_c$  juega el mismo papel que el ángulo límite  $q_0$  juega en la capa mezclada de la teoría de rayos, el ángulo tendido por los rayos atrapados sería  $2q_c$ , y por el mismo tipo de argumento geométrico usado en la sección sobre la capa mezclada la distancia de transición sería  $H / (2q_c)$ .

Las aproximaciones, quizás demasiado amplias, aplicadas a los campos acústicos obtenidos por el método de imágenes, han dado resultados similares a los basados en la teoría de rayos. Una explicación más cuantitativa y exacta del método de imágenes daría resultados más refinados.

### 9. Circuito para el Móvil para Levantamiento Topográfico en Márgenes Acuáticos

El circuito empleado para realizar las pruebas de medición de distancia, mediante ultrasonidos se muestra en la **Figura 8**.

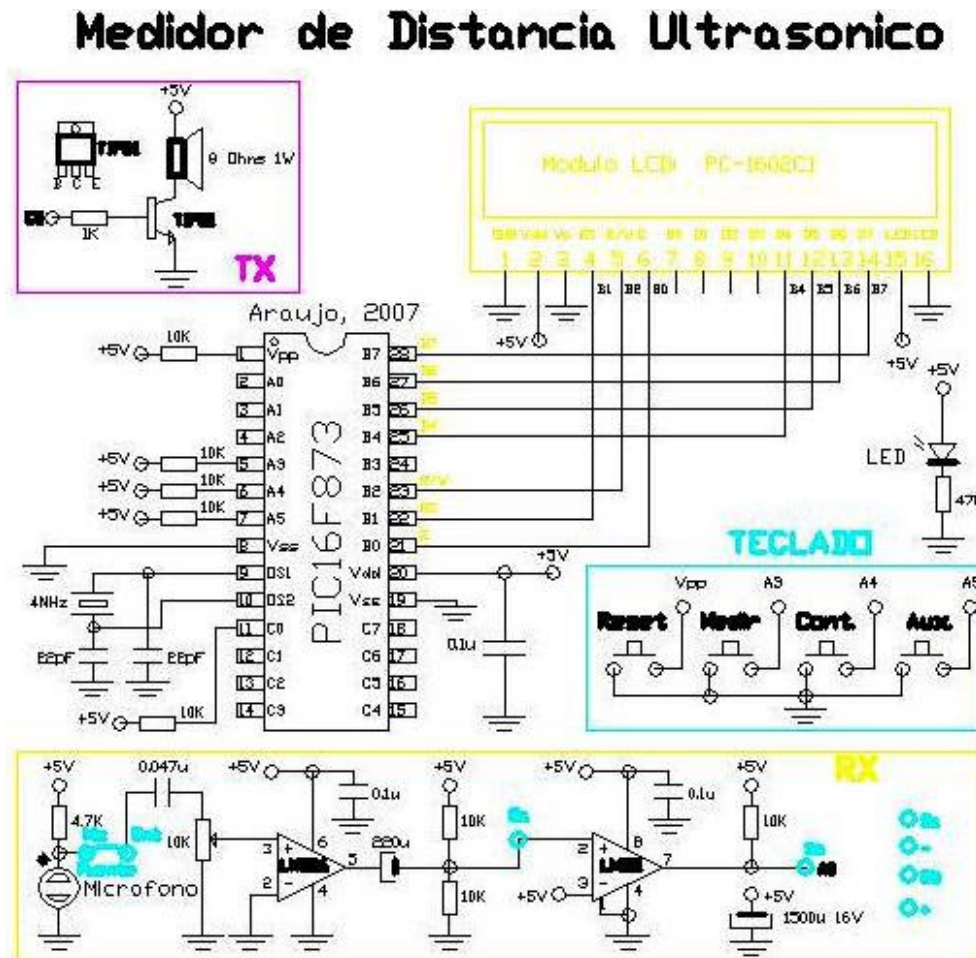


Figura 8: Medidor de distancia ultrasónico.

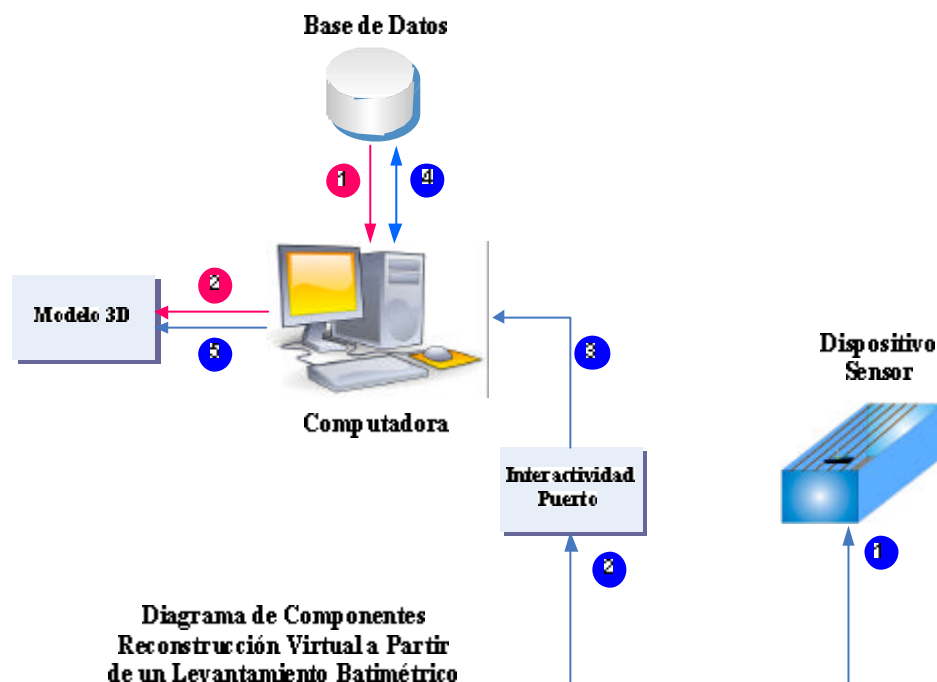
## 10. Resultados.

Después de lo expuesto en el Planteamiento del Problema, se propone crear un sistema que lleve a cabo la reconstrucción virtual a partir de un Levantamiento Batimétrico.

Partiendo de dos métodos para la obtención de datos, a partir de los cuales se realizará la reconstrucción. El primer método consiste en la captura de Datos por un dispositivo (Hardware); mientras que el segundo método parte de los datos almacenados dentro del sistema para generar la Reconstrucción Virtual.

Para cumplir con el primer método planteado de obtención de datos, las pruebas se llevarán a cabo en un medio controlado, lo cual quiere decir que se debe conocer previamente ciertas características, como lo son las dimensiones (profundidades y superficie) del área, este medio representará a escala una presa o lago.

En el siguiente diagrama (**Figura 9**) se muestran los componentes generales del Sistema:



**Figura 9:** Diagrama de Componentes del Sistema.

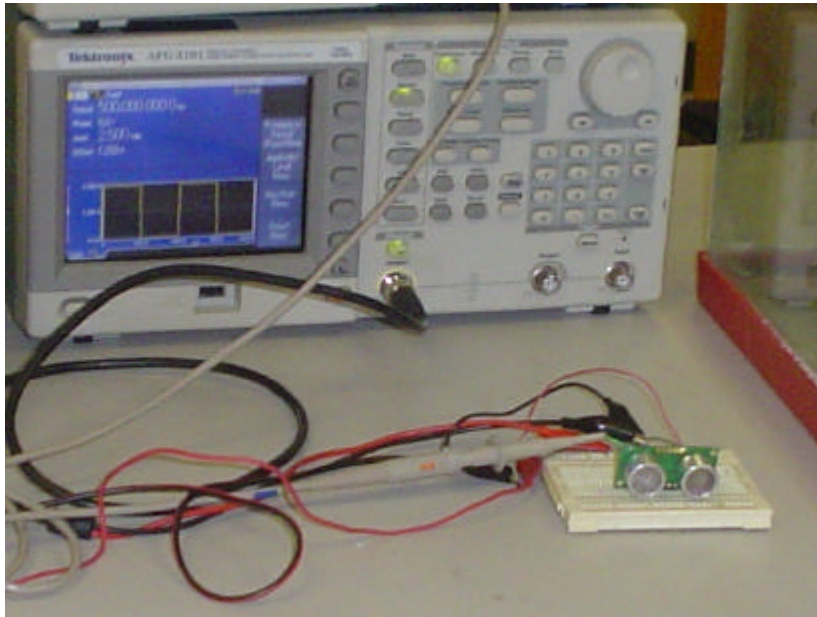
### 10.1 Pruebas con el Sensor.

El sensor ocupado para la adquisición de señales es el **SRF05** en el **modo 1**. El **SRF05**, posee **2** transductores, el **emisor** por el cual se manda un tren de pulsos a una frecuencia de **40KHz** para después convertirlo en energía sonora (ultrasónica), y el **receptor** que por su parte recibirá el mismo tren de pulsos a la frecuencia indicada. Las bondades que ofrece éste dispositivo con acoplamiento integrado es la simplicidad para su manipulación, la cual se puede observar en el diagrama de tiempos de la **Figura 8**.

### 10.2 Señales

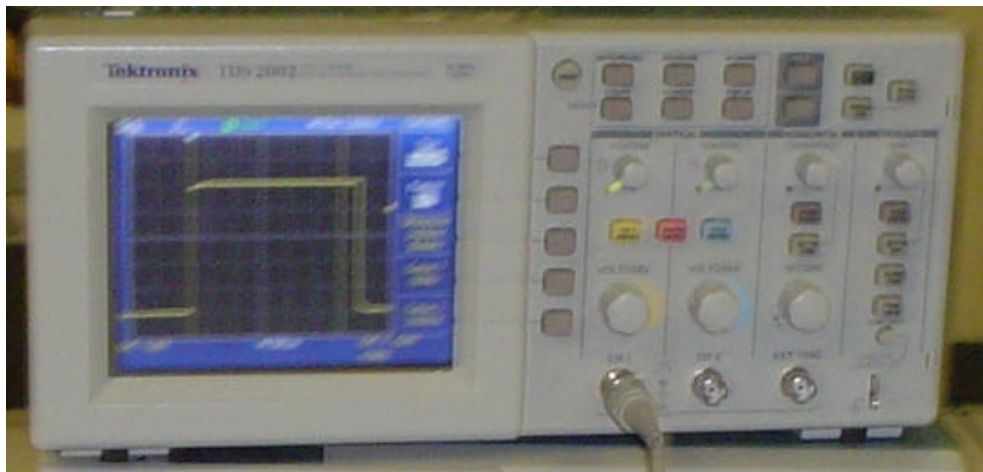
Éste diagrama nos indica que para habilitar la circuitería de acoplamiento, es necesario aplicar un pulso en alto en la terminal de disparo con duración aproximada de **10µs** para iniciar una nueva medición (envió del tren de 8 pulsos a **40KHz**), para después medir el ancho del eco (respuesta) proporcional a la distancia recorrida por la energía sonora ultrasónica, no mayor a **30ms**.

La **Figura 10** permite observar la se1al cuadrada que sirvi3 de disparo para el sensor.



**Figura 10:** Generador de Funciones enviando se1al de disparo al Sensor.

En el Osciloscopio podemos observar la respuesta que nos entrega el sensor cada vez que es detectado o no un objeto (**Figura 11**).



**Figura 11:** Osciloscopio que muestra la se1al de eco, del sensor.

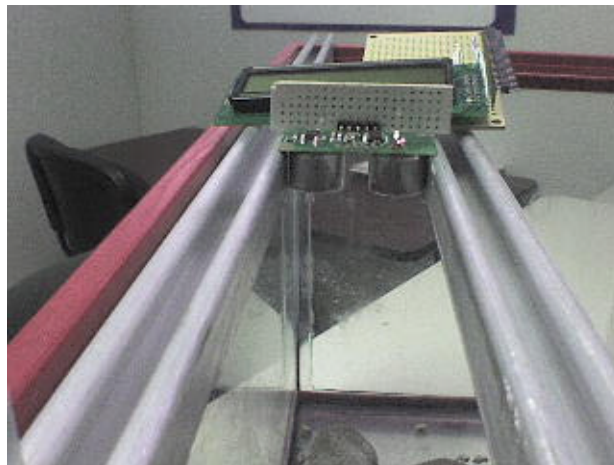
El periodo de la se1al de eco indica que tan cerca o lejos se encuentra el objeto, ya que esta var3a en funci3n del tiempo en que tarda en regresar el sonido.

Como se indico en el An3lisis el medio en el que se realizaron las pruebas fue una pecera, que representa un fondo acu3tico real, y como se observa en la **Figura 12**, se recreo un relieve con piedras que tuviera diferentes niveles y se pudiera observar diferentes distancias, al momento de ir recorriendo con el sensor.



**Figura 12:** Maqueta de un medio real.

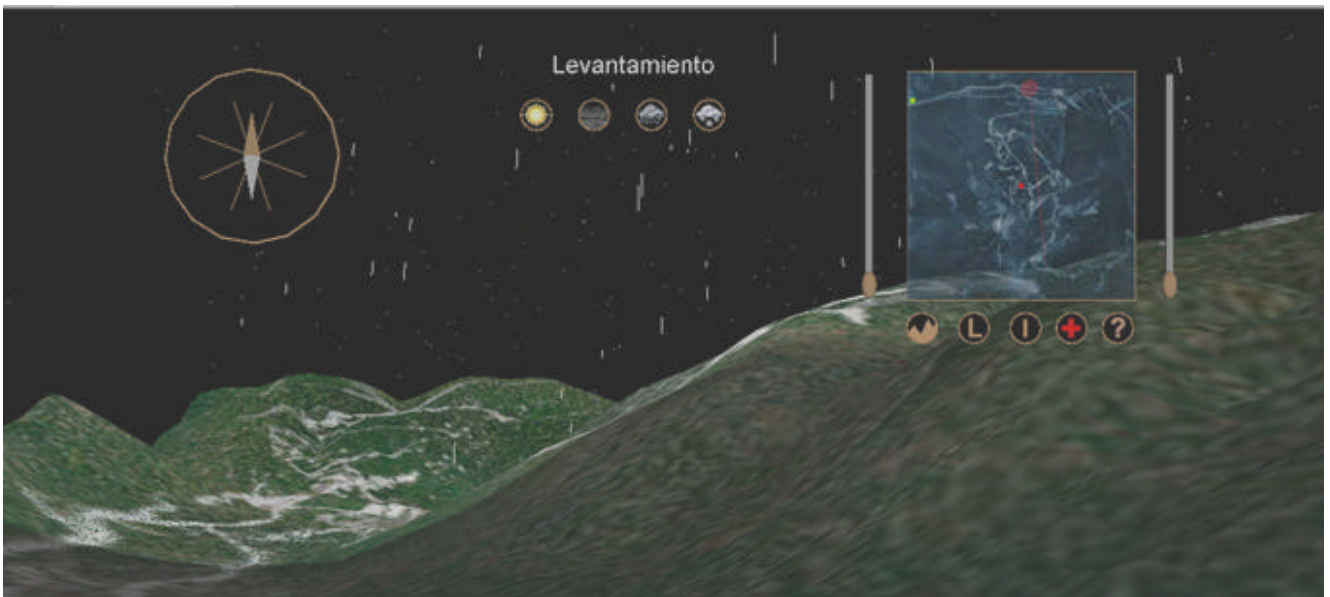
El sistema guía esta conformado por unos rieles en los que se desplazará el sensor para que de esta forma se tengan mediciones precisas a lo largo y ancho del medio (**Figura 13**).



**Figura 13:** Rieles con Sensor montado.

El programa de control se basó en el microcontrolador de Microchip **pic16f873**, debido al anexo de la unidad **USART** (Receptor Transmisor Síncrono Asíncrono Universal Direccional) en su arquitectura, el cual permite hacer la comunicación con cualquier PC con interfaz Serial, además de la API rtx que nos proporciona Java para la comunicación con el puerto serie.

Se realizaron ejemplos en VRML para el modelado de la reconstrucción de un levantamiento.



**Figura 14:** Ejemplo de graficación en VRML

El código fuente para la elaboración del modelado 3D se muestra en el [Programa 1](#).

```
#VRML V2.0 utf8
PROTO Brown1 [] { Material { diffuseColor 0.72 0.56 0.38 shininess 0.4 transparency 0 } }
PROTO Brown2 [] { Material { diffuseColor 0.72 0.56 0.38 emissiveColor 0.4 0.4 0.4 shininess 0.4
transparency 0 } }
PROTO Brown3 [] { Material { diffuseColor 0.72 0.56 0.38 emissiveColor 0.72 0.56 0.38 shininess 0.4
transparency 0 } }
PROTO Compass2d [
eventIn SFVec3f transxyz
eventIn SFRotation rotxyz]
{
  Transform
  {
    set_translation IS transxyz
    set_rotation IS rotxyz
    children [
      DEF needles Transform {
        children [
          Transform {
            rotation 0 0 1 1.5707963267948966192
            children Shape {
              appearance Appearance { material Material { diffuseColor 0.8 0.8 0.8 } }
              geometry DEF arrow IndexedFaceSet {
                ccw TRUE
                solid TRUE
                coord Coordinate { point [ 0 2 0, -8 0 0, 0 -2 0, 0 0 1 ] }
                coordIndex [0,1,2, -1,3,1,2,-1,0,1,3,-1]
              }
            }
          }
          Transform {
            rotation 0 0 -1 1.5707963267948966192
            children Shape {
              appearance Appearance { material Brown1{} }
              geometry USE arrow
            }
          }
        ]
      }
    ]
  }
}
```

**Programa 1:** Código fuente para la graficación 3D en VRML (Parte 1/14).

```

}
}
PROTO HUD [
  exposedField SFVec3f size 3.4e38 3.4e38 3.4e38
  exposedField SFVec3f translation 0.0 0.0 -20
  exposedField SFRotation rotation 0 0 0
  field MFNode children []
]
{
  Group {
    children [
      DEF UserPosition ProximitySensor {
        center 0 0 0
        size IS size
      }
      DEF HUD Transform
      {
        children [
          Transform {
            children [
              Transform {
                translation IS translation
                rotation IS rotation
                children IS children
              }
            ]
          }
        ]
      }
    ]
  }
  ROUTE UserPosition.position_changed TO HUD.translation
  ROUTE UserPosition.orientation_changed TO HUD.rotation
}

PROTO Icon [
  field MFString url []
]
{
  Transform {
    rotation 1 0 0 1.5707963267948966192
    translation 0 -0.01 0
    children [
      Transform {
        translation 0 -0.01 0
        children Shape {
          appearance Appearance { material Brown1{}}
          geometry Cylinder { height 0.01 radius 0.28 }
        }
      }
      Transform {
        children Shape {
          appearance Appearance { material Brown1{}}
          texture ImageTexture { url IS url }
        }
        geometry Cylinder { height 0.01 radius 0.25 }
      }
    ]
  }
}

PROTO MapHud [
  eventOut SFVec3f value_changed
  eventOut SFVec3f value_changed2
  field SFVec3f value 0 0 0
  eventInSFVec3f set_value
  field SFString description ""
  exposedField MFNode maptsv Shape {
    appearance Appearance {

```

Programa 1: Código fuente para la graficación 3D en VRML (Parte 2/14).

```

textureTransform TextureTransform { scale -1 1 }
material Material {
    diffuseColor 1 1 1
    shininess 0.4
    transparency 0.6
}
geometry Box { size 4 4 0.01 }
}

exposedField MFNode Lineframe Transform {
    translation -2 2.01 0
    children Shape {
        appearance Appearance {material Brown3{}}
        geometry IndexedLineSet {
            coord Coordinate { point [ 0 0.01 0, 4.02 0.01 0, 4.02 -4.02 0, 0 -4.02 0 ] }
            coordIndex [ 0, 1, 2, 3, 0, -1]
        }
    }
}

exposedField MFNode boulem Transform {
    children Shape {
        appearance Appearance { material Material {
            diffuseColor 1 0 0
            shininess 0.4
            transparency 0
            emissiveColor 1 0 0
        }
    }
    geometry Sphere { radius 0.05 }
}
}
]
{
Group {
    children [
    DEF TouchMap TouchSensor {}
    Transform {
        #translation 0 0.5 0
        children IS maptsv
    }
    Transform {
        #translation 0 0.5 0
        children IS Lineframe
    }
    Transform {
        children [
        DEF SensorMap PlaneSensor {
            offset 0 0 0
            autoOffset TRUE
        }
    ]
    Transform { children IS maptsv }
    DEF BOULEM Transform { children IS boulem }
    ]}]
}

DEF ScriptMap Script {
    eventOut SFVec3f value_changed IS value_changed
    eventOut SFVec3f value_changed2 IS value_changed2
    eventOut SFVec3f translation_changed

    eventIn SFBool setOver
    eventInSFVec3f set_translation
    eventInSFVec3f set_value IS set_value

    field SFVec3f value IS value
    field SFString description IS description

    url [

```

Programa 1: Código fuente para la graficación 3D en VRML (Parte 3/14).



```

"vrmlscript:function set_value(val, time){
    value = val;
    translation_changed = value;
    value_changed = value;
}
function set_translation(val, time) {
    var a;
    value = val;
    if (value[0] < -2.0) { value[0] = -2.0; }
    if (value[0] > 2.0) { value[0] = 2.0; }
    if (value[1] < -2.0) { value[1] = -2.0; }
    if (value[1] > 2.0) { value[1] = 2.0; }

    translation_changed = value;
    value_changed[0] = - (2747 * value[0] / 4);
    a = value[1];
    value_changed[1] = 0;
    value_changed[2] = - (3576 * a / 4);
    value_changed2[0] = (value[0] * 2747 / 4);
    value_changed2[1] = -141;
    value_changed2[2] = - (value[1] * 3576 / 4);
}
function initialize(val, time){
    translation_changed = value;
}
function setOver(value,time){
    is_over = value;
    if(value){ Browser.setDescription(description);} else { Browser.setDescription('');}
}
}
]
}
ROUTE TouchMap.isOver TO ScriptMap.setOver
ROUTE SensorMap.translation_changed TO ScriptMap.set_translation
ROUTE ScriptMap.translation_changed TO BOULEM.set_translation
ROUTE ScriptMap.translation_changed TO SensorMap.set_offset
}

PROTO Slider [
eventOut      SFFloat value_changed
field         SFFloat value 0.5
eventIn       SFFloat set_value
field         SFString description "slider"
exposedField  MFNode thumb Shape {

    geometry Sphere { radius 0.07 }
    appearance Appearance { material Brown1{}}
}

exposedField MFNode slide Shape {
    geometry Box { size 0.06 1.101 0.01 }
    appearance Appearance {
        material Material {
            ambientIntensity 0.25
            diffuseColor      0.6 0.6 0.6
            shininess 1
        }
    }
}
}
]
{
Group {
    children [
    DEF TOUCH TouchSensor {}
    Transform {
        #translation 0 0.5 0
        children IS slide
    }

    Transform {
        children [

```

Programa 1: Código fuente para la graficación 3D en VRML (Parte 4/14).

```

    DEF SENSOR PlaneSensor {
        minPosition 0. -0.5
        maxPosition 0. 0.5
        offset      0 0 0
        autoOffset  TRUE
    }
    DEF THUMB Transform {
        children IS thumb
    }
    ]
}
]
}

DEF ScriptSlider Script {
    eventOut SFFloat value_changed IS value_changed
    eventOut SFVec3f translation_changed

    eventIn SFBool setOver
    eventInSFVec3f set_translation
    eventInSFFloat set_value IS set_value

    field SFFloat value IS value
    field SFString description IS description

    url ["vrmlscript:function set_value(val, time){
    value = val[1];
    if (value < 0.0) { value = 0.0; }
    if (value > 1.0) { value = 1.0; }

    translation_changed[1] = value;
    value_changed = value;
    }
    function set_translation(val, time) {
        translation_changed[0] = 0;
        translation_changed[1] = val[1];
        value_changed = val[1]+0.5;
    }
    function initialize(val, time){
        translation_changed[0] = 0;
        translation_changed[1] = value-0.5;
    }
    function shutdown(){ }
    function setOver(value,time){
        is_over = value;
        if(value){Browser.setDescription(description);}
        else {Browser.setDescription('');}
    }
    }"
    ]
}
ROUTE TOUCH.isOver TO ScriptSlider.setOver
ROUTE SENSOR.translation_changed TO ScriptSlider.set_translation
ROUTE ScriptSlider.translation_changed TO THUMB.set_translation
ROUTE ScriptSlider.translation_changed TO SENSOR.set_offset
}
PROTO Redt [] { Material {
    diffuseColor 0 0 1
    emissiveColor 0 0 1
    shininess 0.4
    transparency 0.85
}
}

PROTO toggleProto [
    field MFString url []
    eventOut SFBool toggleResult
    ]
{

```

Programa 1: Código fuente para la graficación 3D en VRML (Parte 5/14).

```

DEF toggle Transform {
  rotation 1 0 0 1.5707963267948966192
  translation 0 -0.01 0

  children
  [
    DEF toggleOnOff_0 Script {
      eventOut SFBool      resultBool IS toggleResult
      eventOut SFTime      resultOn
      eventOut SFTime      resultOff
      eventOut SFCOLOR     resultColor
      eventIn SFTime       startTime
      eventIn SFBool       setOver
      field SFBool         state FALSE
      field SFString       description IS description

      url      "vrmlscript:function flipState(value, time)
      {
        state = !state;
        resultBool = state;

        if (state) {
          resultOn = time;
          resultColor = new SFCOLOR(1,0,0);
        }
        else {
          resultOff = time;
          resultColor = new SFCOLOR(0.72,0.56,0.38);
        }
      }
      function startTime(value, time)
      {
        flipState(value, time);
      }
      function setOver(value,time){
        is_over = value;
        if(value){ Browser.setDescription(description);} else { Browser.setDescription(''); }
      }
    }
  ]
  DEF toggleHandle_1 Transform {
    children
    [
      DEF touchSensorTrigger TouchSensor {}
      Group {
        children [
          Transform {
            translation 0 -0.01 0
            children Shape {
              appearance Appearance { material DEF BaseToggle Material {
                diffuseColor 0.72 0.56 0.38 } }
              geometry Cylinder { height 0.01 radius 0.28 }
            }
          }
          Transform {
            children Shape {
              appearance Appearance { material Material { diffuseColor
                0.72 0.56 0.38 } }
              texture ImageTexture { url IS url }
              geometry Cylinder { height 0.01 radius 0.25 }
            }
          }
        ]
      }
    ]
  }
]
}

```

**Programa 1:** Código fuente para la graficación 3D en VRML (Parte 6/14).

```

ROUTE touchSensorTrigger.isOver TO toggleOnOff_0.setOver
ROUTE touchSensorTrigger.touchTime TO toggleOnOff_0.startTime
ROUTE toggleOnOff_0.resultColor TO BaseToggle.set_diffuseColor
}

DEF NavInfo01 NavigationInfo {
  type "ANY"
  avatarSize [0.25, 100, 10]
  headlight TRUE
  speed 35
  visibilityLimit 0
}
DEF Omni01 PointLight {
  intensity 1
  color 0.9725 0.9725 0.8824
  location 810.2 1709 -1387
  on TRUE
  radius 2000
}
DEF Bkg0 Background {skyColor [0 0 0]}
DEF Bkg3 Background {
  skyAngle [ 1.05, 1.57 ]
  skyColor [ 0 1 0.2, 0.1 0.1 0.8, 0.2 0.2 1 ]
}
DEF Bkg2 Background {
  skyAngle [ 0.265, 0.7472, 1.3, 1.5708 ]
  skyColor [ 0.45 0.47 0.43, 0.35 0.4 0.5, 0.35 0.45 0.5, 0.7 0.65 0.75, 0.3 0.3 0.35 ]
}
DEF Bkg1 Background {
  skyAngle [ 0.8 1.57 2.3 ]
  skyColor [
0.21 0.18 0.66
0.2 0.44 0.85
0.51 0.81 0.95
0.77 0.8 0.82
]
}
DEF FogTsv Fog {
  color 1 1 1
  fogType "EXPONENTIAL"
  visibilityRange 0
}
DEF Fog2 Fog {
  color 0.8 0.8 1
  fogType "EXPONENTIAL"
  visibilityRange 22500
}

DEF Px11 ProximitySensor { center 0 0 0 size 2682.9 7000 3558 }

DEF monde Transform {
  translation 0 0 0
  children [
    DEF socle Transform {
      translation 0 -141.9 0
      children Shape {
        appearance Appearance { material Brown1{} }
        geometry Box { size 2682.9 50 3558 }
      }
    }
    DEF Plane Transform {
      translation 0 -141.9 0
      children Collision { collide FALSE children Shape {
        appearance Appearance { material Redt{} }
        geometry Box { size 2682.9 0.1 3558 }
      }
    }
  ]
}

```

**Programa 1:** Código fuente para la graficación 3D en VRML (Parte 7/14).



```

DEF Reduser Transform {
  translation 0 0 0
  children [
    Transform {
      children Shape {
        appearance Appearance {
          material Material { emissiveColor 1 0 0 diffuseColor 1 0 0 }
        }
        geometry IndexedLineSet {
          coord Coordinate { point [ 0 -141 0, 0 1100 0 ] }
          coordIndex [ 0, 1, -1]
        }
      }
    }
  ]
  Transform {
    translation 0 1100 0
    children Shape {
      appearance Appearance { material Material {
        diffuseColor 0.7 0 0
        shininess 0
        transparency 0
        emissiveColor 0.5 0 0
      }
    }
    geometry Sphere { radius 20 }
  }
}

DEF S Script {
eventIn      SFBool c1
eventIn      SFVec3f set_value
eventOut     SFVec3f value_changed

  url "vrmlscript:function set_value(val, time){
    value_changed[0] = val[0]*4 / 2747;
    value_changed[1] = - (val[2]*4 / 3576);
    value_changed[2] = 0;
  }
  function initialize(){}
  function shutdown(){}"
}

DEF ScriptC2d Script {
eventIn SFRotation set_value
eventOut SFRotation value_changed
url ["vrmlscript:function set_value(val, time){
  value_changed = val;
  value_changed[0] = 0;
  value_changed[1] = 0;
  value_changed = value_changed.inverse();
}
function initialize(){}
function shutdown(){}"
]
}

HUD {
  translation 0.1 0.60 -1
  children [
    Transform {
      translation 0.3 -0.4 0
      scale 0.07 0.07 0.07
      children [
        Transform {
          translation 0 0 -0.1
          children DEF Minimap MapHud {
            value 0 0 0
          }
        }
      ]
    }
  ]
}

```

**Programa 1:** Código fuente para la graficación 3D en VRML (Parte 9/14).

```

        description "H.U.D. - MiniMap : Drag the red point to set a reference..."
    }
}
Anchor {
    url ["help.html"]
    description "Get help on this world..."
    children Transform { translation 1.6 -2.5 0.1 children Icon { url "help.jpg" }}
}
Anchor {
    url ["medic.html"]
    description "Medical Information in the Taos Ski Valley"
    children Transform { translation 0.8 -2.5 0.1 children Icon { url "cross.jpg" }}
}
DEF Altitude Transform {
    children Shape {
        appearance Appearance {
            material Material {
                diffuseColor 1 1 1
                shininess 1
            }
        }
        geometry DEF Alt1 Text {
            fontStyle FontStyle {
                family "Arial"
                justify "MIDDLE"
            }
            string "Araujo 2007"
        }
    }
    translation -8 7 -15
}

Transform { children [
    DEF SW Switch {
        whichChoice 0 # set by Script
        choice [
        ]
    }
    DEF TSW TouchSensor {}
    DEF SCR Script { # Switches the choice
        eventIn SFTime touchTime
        eventOut SFInt32 whichChoice
        url ["vrlmscript:
        function initialize() {
            whichChoice = 0;
        }
        function touchTime( value, time) {
            if ( whichChoice == 1 ) whichChoice = 0;
            else ++whichChoice;
        }"]
    }
]}
Transform {
    translation -2.7 0 0
    scale 2 3.5 1
    children DEF SLIDER_ALT Slider {
        value 0
        description "Altitude Plane Slider"
    }
}

Transform {
    translation 2.55 0 0
    scale 2 3.5 1
    children DEF SLIDER_LIGHT Slider {
        value 0
        description "Light Slider"
    }
}
}

```

**Programa 1:** Código fuente para la graficación 3D en VRML (Parte 10/14).

```

DEF userloc Transform { translation -2 2 0 children Shape {
  appearance Appearance { material Material {
    diffuseColor 1 0 0
    shininess 0.4
    transparency 0
    emissiveColor 0 1 0
  }
  geometry Sphere { radius 0.05 }
}
}
Transform {
  translation -13.5 0.50 0
  scale 0.075 0.1 0.075
  children DEF Hcompass Compass2d { rotxyz 0 1 0 0 }
}
DEF socleC2 Transform {
  translation -13.5 0.50 0
  children DEF ssc2 Transform {
    rotation 1 0 0 1.5707963267948966192
    children [
      Transform {
        translation 0 -0.01 0
        children Shape {
          appearance Appearance { material Material { diffuseColor 0 0 0
transparency 0 } texture ImageTexture { url "compass.jpg" }}
          geometry Cylinder { height 0.01 radius 1.5 }
        }
      }
      Transform {
        translation 0 -0.02 0
        children Shape {
          appearance Appearance { material Brown1{}}
          geometry Cylinder { height 0.01 radius 1.54 }
        }
      }
      Transform {
        children [
          Transform {
            children DEF axecompass Shape {
              appearance Appearance { material Brown3{}}
              geometry IndexedLineSet {
                coord Coordinate { point [ -1 0 0, 1 0 0 ] }
                coordIndex [ 0, 1, -1]
              }
            }
          }
          Transform { rotation 0 1 0 0.6872233929727672709 children USE axecompass }
          Transform { rotation 0 1 0 1.5707963267948966192 children USE axecompass }
          Transform { rotation 0 1 0 2.3561944901923449288 children USE axecompass }
        ]
      }
    ]
  }
}
]
}
}
]
}
DEF PlaneInterpolator PositionInterpolator{
  key [0,0.5,1]
  keyValue[
    0 24 0,
    0 450 0,
    0 910 0
  ]
}

```

**Programa 1:** Código fuente para la graficación 3D en VRML (Parte 11/14).



```

children [
  DEF build_snow Script {
    eventOut MFVec3f points_out
    eventOut MFVec3f points_out1
    url "vrmlscript:
      function initialize() {
        points_out = new MFVec3f();
        points_out1 = new MFVec3f();
        for (xx = 0; xx<nbrx; xx++) {
          for (zz = 0; zz<nbrz; zz++) {
            px = -(xdim/2) + Math.random()*xdim;
            py1 = py - (Math.random()*alty);
            pz = -(zdim/2) + Math.random()*zdim;

            points_out[p] = new SFVec3f(px,py1,pz);
            points_out1[p] = new SFVec3f(px,py1,pz);

            points_out[p+((nbrx*nbrz))] = new SFVec3f(px,py1-py,pz);
            p++;
          }
        }
      }
    }
  }
}

Group {
  children Shape {
    appearance Appearance {
      material Material {
        diffuseColor 0.8 0.8 0.9
        emissiveColor 0.8 0.8 0.9
      }
    }
    geometry DEF snowflakes PointSet {
      coord DEF snowcoord Coordinate { }
    }
  }
}

DEF snowfall CoordinateInterpolator {
  key [ 0, 1 ]
  keyValue []
}

DEF Tsnow TimeSensor {
  loop TRUE
}

]
ROUTE build_snow.points_out1 TO snowcoord.set_point
ROUTE build_snow.points_out TO snowfall.set_keyValue
ROUTE Tsnow.fraction_changed TO snowfall.set_fraction
ROUTE snowfall.value_changed TO snowcoord.set_point
}

DEF Rain Transform {
  translation 0 900 0
  children DEF Crain Transform {
    children [
      DEF build_rain Script {
        eventOut MFVec3f points_out
        eventOut MFVec3f points_out1
        eventOut MFInt32 lines_out
        url "vrmlscript:
          function initialize() {
            points_out = new MFVec3f();
            points_out1 = new MFVec3f();

```

**Programa 1:** Código fuente para la graficación 3D en VRML (Parte 12/14).

```

lines_out = new MFInt32();
for (xx = 0; xx<nbrx; xx++) {
for (zz = 0; zz<nbrz; zz++) {
    px = -(xdim/2) + Math.random()*xdim;
    py1 = py - (Math.random()*alty);
    pz = -(zdim/2) + Math.random()*zdim;
    py2 = py1 - Math.random()*lrain;

    points_out[p] = new SFVec3f(px,py1,pz);
    points_out[p+(nbrx*nbrz)] = new SFVec3f(px,py2,pz);
    points_out1[p] = new SFVec3f(px,py1,pz);
    points_out[p+((nbrx*nbrz)*2)] = new SFVec3f(px,py1-py,pz);
    points_out[p+((nbrx*nbrz)*3)] = new SFVec3f(px,py2-py,pz);
    p++;
}
}
for (li = 0; li<(nbrx*nbrz); li=li+3) {
    lines_out[li] = li;
    lines_out[li+1] = li + (nbrx*nbrz);
    lines_out[li+2] = -1;
}
}
}

Group {
    children      Shape {
        appearance Appearance {
            material      Material {
                diffuseColor  0.8 0.8 0.9
                emissiveColor  0.8 0.8 0.9
            }
        }
        geometry DEF raindrops IndexedLineSet {
            coord DEF raincoord Coordinate { }
        }
    }
}

}
DEF rainfall CoordinateInterpolator {
    key      [ 0, 1 ]
    keyValue []
}
DEF Train TimeSensor {
    loop TRUE
}

]
ROUTE build_rain.points_out1 TO raincoord.set_point
ROUTE build_rain.lines_out TO raindrops.set_coordIndex
ROUTE build_rain.points_out TO rainfall.set_keyValue
ROUTE Train.fraction_changed TO rainfall.set_fraction
ROUTE rainfall.value_changed TO raincoord.set_point
}
}

DEF ScriptAlt Script {
    eventIn SFVec3f set_value
    eventOut MFString Saltitude
    field SFInt32 value 0
    url ["vrmlscript:function set_value(val){
        value = 9085+(val[1]*2927/885);
        Saltitude = new MFString('Altitud del Plano : '+value.toString()+ ' Pies');
    }"
]
}
}

```

**Programa 1:** Código fuente para la graficación 3D en VRML (Parte 13/14).

```

DEF SSnow Script {
    eventIn SFBool isActive
    eventOut MFNode childon
    eventOut MFNode childoff
    eventOut SFBool first

    field MFNode Node USE Csnow
    url "vrmlscript:
function isActive(value) {
if(value) {childon = Node} else {childoff = Node};
first = FALSE;
}"
}

DEF SRain Script {
    eventIn SFBool isActive
    eventOut MFNode childon
    eventOut MFNode childoff
    eventOut SFBool first

    field MFNode Node USE Crain
    url "vrmlscript:
function isActive(value) {
if(value) {childon = Node} else {childoff = Node};
first = FALSE;
}"
}

DEF TT1 TimeSensor {
    loop TRUE
}
DEF TT2 TimeSensor {
    loop TRUE
}

ROUTE TT1.isActive TO SRain.isActive
ROUTE TT2.isActive TO SSnow.isActive
ROUTE SRain.first_changed TO TT1.set_loop
ROUTE SSnow.first_changed TO TT2.set_loop

ROUTE TSW3.toggleResult TO SRain.isActive
ROUTE SRain.childon TO Rain.addChildren
ROUTE SRain.childoff TO Rain.removeChildren

ROUTE TSW4.toggleResult TO SSnow.isActive
ROUTE SSnow.childon TO Snow.addChildren
ROUTE SSnow.childoff TO Snow.removeChildren

ROUTE SLIDER_ALT.value_changed TO PlaneInterpolator.set_fraction
ROUTE PlaneInterpolator.value_changed TO Plane.set_translation
ROUTE PlaneInterpolator.value_changed TO ScriptAlt.set_value
ROUTE ScriptAlt.Saltitude TO Alt1.set_string

ROUTE TSW1.toggleResult TO Bkg1.set_bind
ROUTE TSW2.toggleResult TO Fog2.set_bind
ROUTE TSW3.toggleResult TO Bkg2.set_bind

ROUTE TSW.touchTime TO SCR.touchTime
ROUTE SCR.whichChoice TO SW.whichChoice
ROUTE SCR.whichChoice TO LandSW.whichChoice

ROUTE SLIDER_LIGHT.value_changed TO Omni01.set_intensity

ROUTE Px11.position_changed TO S.set_value
ROUTE Px11.orientation_changed TO ScriptC2d.set_value
ROUTE ScriptC2d.value_changed TO socleC2.set_rotation
ROUTE S.value_changed TO userloc.set_translation
ROUTE Minimap.value_changed2 TO Reduser.set_translation

```

**Programa 1:** Código fuente para la graficación 3D en VRML (Parte 14/14).

## 11. Impacto.

La herramienta desarrollada facilitará la toma de decisiones de los expertos en el tema, para la resolución de los diversos problemas en sus proyectos hidrobiológicos.

## 12. Conclusiones

Parte fundamental del Sistema a desarrollar fue el análisis para entender de que forma un Levantamiento batimétrico se lleva a cabo en la vida real, de lo que se concluye que existen varias etapas especializadas para sacar el resultado final. En el caso del presente proyecto el enfoque va encaminado hacia la captura de datos de profundidades en una determinada área, y la graficación de esta información.

Entre los diferentes medios de comunicación con la computadora analizamos diferentes puertos, pero para efectos prácticos se trabajó con el puerto serial.

Para la parte de software se decidió usar el modelado en VRML con sus funciones establecidas nos auxilió en la generación y graficación del levantamiento de una forma sencilla.

## 13. Referencias.

### 13.1 Internet

- [1] Google - [Equipo de AdWords](http://www.clubdelamar.org/batimetria.htm),  
<http://www.clubdelamar.org/batimetria.htm>
- [2] **Angela María Galeano P.** [agaleano\\_2000@yahoo.com](mailto:agaleano_2000@yahoo.com) 2001  
<http://www.angelagaleano.freesevers.com/>
- [3] M. en C. David Araujo Díaz  
Página con trabajos realizados durante la **Maestría en Ciencias en Ingeniería Eléctrica** opción de **Computación** realizada en el [Centro de Investigación y de Estudios Avanzados \(CINVESTAV\)](#) del [Instituto Politécnico Nacional \(IPN\)](#).  
<http://www.geocities.com/daraujo16/>
- [4] Comparativa de microcontroladores actuales  
<http://server-die.alc.upv.es/asignaturas/LSED/2002-03/Micros/downloads/trabajo.pdf>
- [5] Manual de Microcontrolador 16F873  
[http://www.fimee.ugto.mx/webfimee2006/MANUALES/Manual\\_microcontrolador16F873.pdf](http://www.fimee.ugto.mx/webfimee2006/MANUALES/Manual_microcontrolador16F873.pdf)
- [6] <http://aprender-java.blogspot.com/>
- [7] <http://www.desarrolloweb.com/articulos/561.php>
- [8] <http://es.wikipedia.org/wiki/C++>
- [9] [http://es.wikipedia.org/wiki/Visual\\_Basic](http://es.wikipedia.org/wiki/Visual_Basic)
- [10] <http://www.monografias.com/trabajos43/java-tres-d/java-tres-d.shtml>
- [11] <http://sabia.tic.udc.es/gc/Contenidos%20adicionales/trabajos/3D/x3d/Conociendo%20X3D.htm>
- [12] <http://es.wikipedia.org/wiki/Blender>
- [13] <http://www.revista.unam.mx/vol.2/num2/art2/index.html>

### 13.2 Bibliográficas

- [Toscano 1958] Toscano Ricardo; **“MÉTODOS TOPOGRÁFICOS”**, Ed. Romero, 1958, 930 pp.
- [García 2000] García Márquez Fernando; **“Curso Básico de Topografía planimetría, agrimensura y altimetría”**, PAX México, 1-31 pp.
- [Roger 2002] **“Ingeniería de Software un enfoque práctico”**, Quinta Edición, Edit Mc.Graw Hill.