

SIMULACIÓN EN TIEMPO REAL DE UN RADAR DE BARRIDO HORIZONTAL

Ignacio Otheguy*, Mariana Soriano*, Gustavo Boroni†, Marcelo J. Vénere †,

* Universidad FASTA - INVAP S.E.
F. P. Moreno 1089, 8400 Bariloche, Argentina

† PLADEMA-CNEA
Universidad Nacional del Centro
Pinto 399
7000 Tandil, Argentina
e-mail: venerem@exa.unicen.edu.ar

Key words: Simulador Radar, Geometría Computacional, Simulación Computacional.

Resumen. *En el presente trabajo se describe el desarrollo de un simulador radar de barrido horizontal, con el objeto de utilizarlo en un entorno de realidad virtual para el entrenamiento de personal. La aplicación es radares en barcos, por lo cual los ecos provendrán principalmente de la topografía de la costa, construcciones y otros barcos o boyas. El modelo computacional propuesto es una aproximación razonable del problema real que permite evitar la generación de imágenes incorrectas, independientemente del escenario planteado. Sombras, ecos provenientes del oleaje, efectos meteorológicos e incluso ruido en la señal son representados con gran realismo. De hecho las imágenes radar que se obtienen son superiores a las de la mayor parte de los productos existentes en el mercado. A su vez, un original empleo de técnicas de clasificación geométrica permitió mantener el costo computacional dentro de rangos aceptables para una aplicación de tiempo real. La herramienta implementada forma parte del sistema MELIPAL, actualmente en funcionamiento en la Escuela Nacional de Náutica, dependiente de la Armada Argentina.*

1 INTRODUCCION

Una parte importante de la instrucción de personal que deberá navegar barcos de gran porte es el manejo de un radar moderno, el cual no solo permite detectar blancos en un entorno de varias millas, sino también realizar un seguimiento de los mismos estimando tiempos y distancias mínimas de acercamiento. Normalmente el entrenamiento con estos equipos se realiza sobre simuladores, y de hecho un navegante deberá cumplir un cierto número de horas en un simulador radar aprobado por la Organización Marítima Internacional (OMI) para poder obtener su licencia.



Figura 1: Pantalla de un radar real instalado en un barco que se encuentra en puerto

En la figura 1 se muestra la pantalla de un radar real como el que se quiere simular, en el que se observan los ecos provenientes de las dársenas (el barco se encontraba en puerto), y las distintas funciones disponibles en el mismo. Es común que estos radares se operen casi completamente con un puntero del tipo *trackball*, por lo cual la funcionalidad se visualiza en pantalla como botones o menús. En realidad la pantalla de un radar moderno es un display raster equivalente al de una computadora, sobre el cual se simula el haz de barrido. Además, la mayor parte de la funcionalidad del radar se implementa en forma digital, por lo cual su simulación en una computadora será totalmente realista si se logra simular correctamente la señal proveniente de la antena.

El requerimiento de realismo es que las imágenes generadas sean aproximadamente iguales a las que se ven en el caso real y por lo tanto los simuladores existentes emplean diferentes trucos que no necesariamente tienen un sustento en el proceso físico. Una de las implementaciones relativamente buenas puede verse en Transas¹ por ejemplo. En nuestro caso hemos seguido una estrategia más conservadora en la que el problema real es discretizado minimizando las aproximaciones introducidas, de forma de garantizar un modelo lo más realista posible. Por supuesto, este afán de minimizar las aproximaciones compite con el costo computacional del algoritmo, el cual debe ser optimizado de forma de obtener una aplicación que genere las imágenes en el mismo tiempo que un radar real.

En este trabajo presentamos el desarrollo e implementación de un simulador radar que cumple satisfactoriamente con los requerimientos de realismo y tiempo real y que forma parte del sistema MELIPAL-R, cuya primera versión se encuentra en operación en un esquema de cuatro puestos alumnos y un instructor en instalaciones de la Escuela Nacional de Náutica. En primer lugar se describe el modelo propuesto y las aproximaciones realizadas, a continuación se discute brevemente su implementación computacional y finalmente se presentan los algoritmos con que se consigue obtener el costo deseado.

2 EL PROBLEMA Y SU DISCRETIZACION

El principio de funcionamiento de un radar (ver Cook et al² o Meikle³ por ejemplo) se basa en la escucha de los ecos generados por la emisión de un corto pulso electromagnético. Este pulso debe ser colimado de forma que los ecos provengan mayoritariamente de la dirección que se está observando, sin embargo los pulsos que se logran en la realidad tienen el aspecto que se muestra en la figura 2, donde se observa un lóbulo principal con un cierto ancho en la dirección deseada y una serie de lóbulos laterales. La calidad del radar estará asociada a cuan pequeños son los lóbulos laterales y cuan estrecho es el lóbulo principal, ya que los blancos que se encuentren en las zonas alcanzadas por estos lóbulos podrán generar ecos y es claro que aquellos que no provengan de la dirección principal serán interpretados erróneamente.

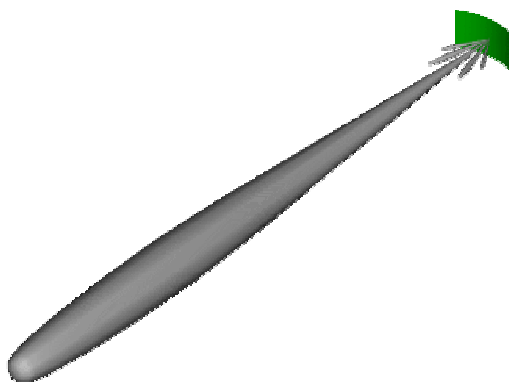


Figura 2: Esquema de un pulso del radar. La zona gris representa la región donde un blanco generará un eco

En el caso que nos interesa, radares en barcos, este pulso es emitido horizontalmente en distintas direcciones, girando la antena a razón de 14 a 20 revoluciones por minuto, con aproximadamente 4000 procesos emisiones-escuchas cada 360 grados.

La discretización natural de este problema puede realizarse considerando al lóbulo como un haz de rayos y a los blancos como un conjunto de polígonos en un espacio tridimensional, tal como se representa en la figura 3. Un proceso de emisión-escucha se transforma entonces en encontrar las intersecciones de este haz de rayos con todos los polígonos, graficando luego en la pantalla del radar

la primera intersección para cada rayo. El costo computacional para este algoritmo sería $O(N_{pol} \cdot N_{ray})$, donde

N_{pol} : Número de polígonos empleados para describir el escenario y los blancos

N_{ray} : Número de rayos utilizados para discretizar el lóbulo.

Normalmente la representación de un escenario tridimensional (topografía de la costa, edificaciones, otros barcos y boyas) se realiza utilizando triángulos y aún para discretizaciones groseras será necesario utilizar del orden de 10^4 a 10^5 triángulos. Si se tiene en cuenta que el cálculo de la intersección entre una recta y un triángulo en el espacio requiere más de cincuenta operaciones de punto flotante, además de cuatro raíces cuadradas, rápidamente se llega a tiempos de cálculo incompatibles con una aplicación en tiempo real, al menos en computadoras estándar.

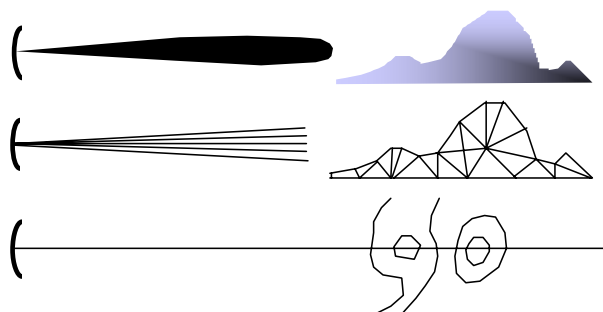


Figura 3: Discretización y aproximación del problema.

Por este motivo hemos propuesto un modelo en donde la tercera dimensión es condensada y el lóbulo es considerado como un único rayo paralelo a la superficie. Para eliminar la tercera dimensión y no perder la información de las pendientes de terreno, representamos la topografía y construcciones mediante líneas de nivel, recurriendo a la interpolación de alturas entre las mismas. De esta forma se obtienen imágenes de la línea de costa muy similares a la que se observa en un radar real y el problema se redujo a evaluar intersecciones entre una semi-recta y el conjunto de segmentos que definen las líneas de nivel.

Para definir la cantidad de puntos a interpolar entre dos líneas hemos propuesto una dependencia proporcional a la diferencia de altura entre las mismas; de esta forma un acantilado se verá nítidamente, mientras que una playa aparecerá en forma difusa. Por otro lado la intensidad de estos ecos en la pantalla es una función del blanco (vegetación produce un eco muy distinto que un edificio por ejemplo) y para considerar también este efecto hemos incorporado una propiedad “intensidad” en cada segmento utilizado para definir las líneas de nivel.

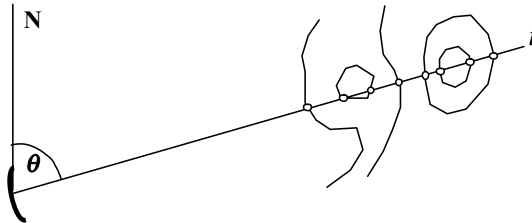


Figura 4: Intersecciones detectadas para una dada posición del haz de barrido.

En la figura 4 se muestran las intersecciones detectadas para una dada posición del haz de barrido. Para encontrar los puntos que deberán ser pintados en la pantalla del radar resulta conveniente escribir la ecuación de la recta en forma paramétrica y ordenar las intersecciones en orden creciente del parámetro t ,

$$\begin{aligned} X(t) &= \cos(\theta).t + Px \\ Y(t) &= \sin(\theta).t + Py \end{aligned} \tag{1}$$

(Px,Py) : Posición del barco

θ : Angulo donde apunta la antena

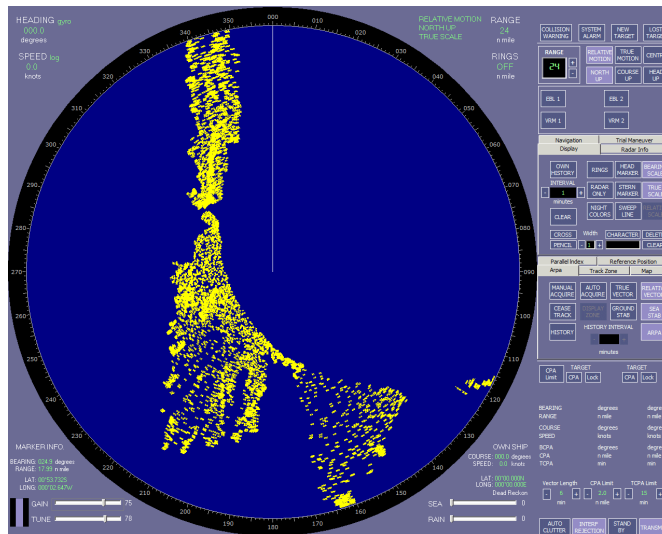


Figura 5: Imagen radar virtual generada para la costa en la zona del Golfo San Matías.

De esta forma el algoritmo de pintado de pixeles en la pantalla del radar consiste en recorrer la lista ordenada de intersecciones t_i , y en el caso de que la altura de la misma sea mayor que la anterior t_{i-1} , se pintará al pixel con la intensidad indicada en el segmento y se generan n puntos adicionales entre t_{i-1} y t_i , siendo n proporcional a la diferencia de alturas. En el caso que la altura en t_i sea menor que en el punto anterior, no se pinta el pixel ni se interpolan nuevos puntos, pero la lista debe seguirse recorriendo para considerar el caso que una intersección más lejana posea una altura suficiente para ser detectada.

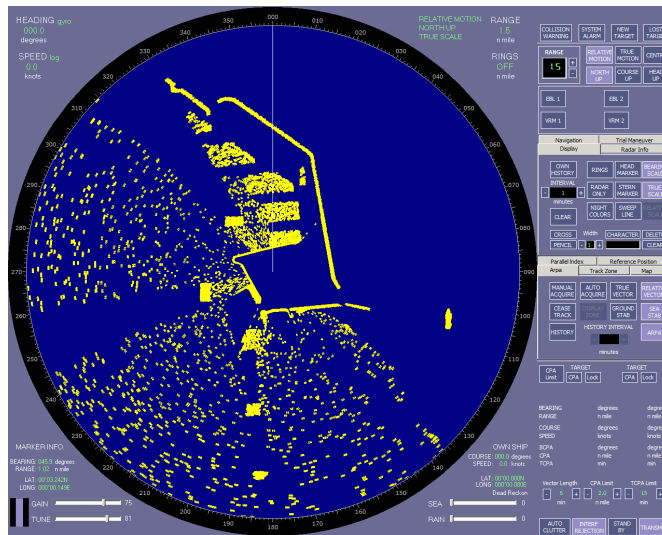


Figura 6: Imagen radar virtual de la zona del puerto de Buenos Aires

Resumiendo entonces, los blancos (móviles o fijos) son representados como un conjunto de segmentos. Cada segmento tiene asociado una altura y una intensidad. Esta intensidad es en general un valor positivo, por lo que valores negativos pueden reservarse para indicar segmentos que no ocultan los objetos que están detrás, como puede ser el caso de puentes.

La figura 5 muestra una imagen radar virtual generada con esta metodología correspondiente a una zona cerca del Golfo San Matías. Obsérvese como se visualizan en forma diferente las zonas de playa al sur comparadas con las costas más abruptas al oeste. La figura 6 muestra otra imagen correspondiente al puerto de Buenos Aires donde además de las dársenas se pueden ver algunos edificios y la sombra que generan los mismos. También se pueden ver otros dos navíos, uno dentro y otro fuera del puerto.

3 OPTIMIZACION DE PERFORMANCE

. Como ya se mencionó, un radar como el que nos interesa debe poder girar a 14 revoluciones por minuto como mínimo, y teniendo en cuenta los 4000 procesos de emisión cada 360 grados, se necesitaran evaluar aproximadamente 1000 de estos procesos por segundo. Si bien con la aproximación realizada al eliminar la tercera dimensión se reduce notablemente el costo computacional, en caso de escenarios muy complejos pueden ser necesarios miles de segmentos y la performance del algoritmo puede no ser la requerida. De hecho en las pruebas iniciales realizadas sobre los escenarios más complejos, procesando sobre equipos Pentium III de 1.0 GHz, el rendimiento obtenido era de apenas cinco revoluciones por minuto.

Del análisis realizado se detectó que el costo computacional principal estaba asociado al cálculo de las intersecciones rayo-segmentos, ya que para cada rayo se analizaban la totalidad de los segmentos. Se optimizó en lo posible esta operación, dejando incluso precalculados aquellos coeficientes que no varían para cada una de las 4000 semirectas, pero aún así se estaba lejos de los tiempos requeridos.

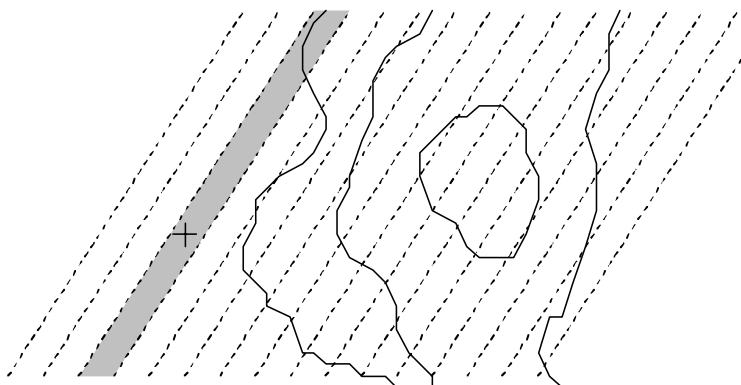


Figura 7: Clasificación de los segmentos en bandas equiespaciadas

La solución clásica a este tipo de problemas es recurrir a técnicas de clasificación geométrica (ver Dari y Vénere⁵ por ejemplo) utilizando estructuras auxiliares como quadtrees o grillas regulares. Sin embargo el caso que nos interesa presenta particularidades que vuelven poco adecuados a todos estos métodos.

Dado que se buscan todas las intersecciones de segmentos con una semi-recta, el método de clasificación deberá permitir encontrar fácilmente los segmentos que se ubican en una banda que contiene a la semi-recta. Esto no resulta sencillo utilizando una grilla y mucho menos con un quadtree. La estructura ideal sería clasificar en bandas paralelas a la semi-recta, tal como se muestra en la figura 7. La implementación más sencilla es tomar un ancho de banda fijo con un tamaño tal que ninguna tenga más de una cierta cantidad dada de segmentos. Esta condición puede resultar muy

exigente para casos en que una gran cantidad de segmentos están en línea, en cuyo caso debería utilizarse bandas de tamaño variable (será necesario un árbol binario para encontrar la banda que corresponde a la posición del barco).

En nuestro caso se realizó una implementación con bandas fijas en 360 direcciones diferentes, consiguiéndose una mejora notable en el tiempo de ejecución, permitiendo cumplir holgadamente con el requerimiento de más de 14 revoluciones por minuto, aún para los escenarios más complejos.

El cálculo de las intersecciones para un dado rayo se realiza entonces en tres pasos: en primer lugar, dado el ángulo de la semirecta se selecciona la clasificación en bandas correspondiente; luego dada la posición del barco se evalúa cual es la banda que le corresponde y se recuperan los segmentos asociados a la misma y a la vecina (si se realizaran 4000 clasificaciones no sería necesario tomar la vecina inmediata, pero eleva considerablemente el costo de almacenamiento); finalmente se calculan las intersecciones entre la semi-recta y los segmentos.

El simulador debe barrer la pantalla a un ritmo constante independientemente de la complejidad del escenario, por lo cual el cálculo de cada una de las 4000 direcciones se realiza con un proceso independiente que se ejecuta a intervalos regulados por el proceso principal del radar.

4 SISTEMA DE ENTRENAMIENTO

Para que el simulador radar desarrollado pueda utilizarse con fines de entrenamiento, debe también simularse que el mismo se encuentra sobre un barco que se desplaza en un dado escenario y con la presencia de otros barcos, también en movimiento. El sistema completo, llamado MELIPAL-R y esquematizado en la figura 8, comprende varios puestos de entrenamiento distribuidos en una LAN monitoreados por un instructor, donde cada puesto está formado por tres equipos: el radar, los controles del barco y la cartografía.

La computadora que presenta los controles del barco (básicamente el timón y el control de potencia) realiza también una simulación del desplazamiento del barco considerando modelos simplificados para embarcaciones que van desde supertanques, transportador de contenedores a embarcaciones livianas). La cartografía es presentada con un sistema comercial estándar el cual permite además ingresar una señal de equipo GPS (simulada también por el sistema) para una ubicación automática sobre el mapa.

Por último el puesto del instructor dispone de herramientas para seleccionar escenarios, asignar embarcaciones, crear situaciones, monitorear desplazamientos e incluso visualizar en tiempo real, sobre un equipo auxiliar, el estado de cualquier radar de los puestos de entrenamiento a efectos de analizar si el mismo está siendo utilizado correctamente. También cuenta con la posibilidad de grabar completamente un ejercicio para una posterior reproducción y estudio.

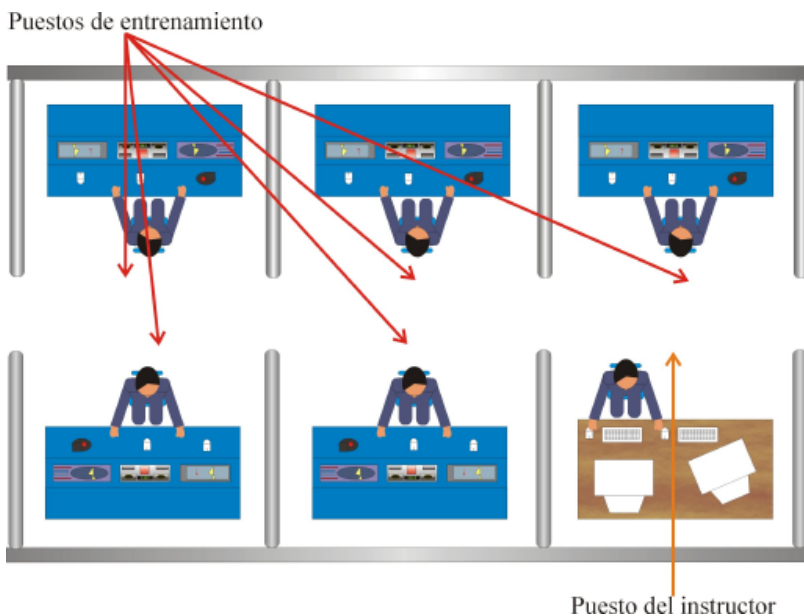


Figura 8: Esquema físico del sistema MELIPAL-R

5 CONCLUSIONES

Se presentó un desarrollo novedoso del modelo computacional que permite simular un radar de barrido horizontal. Las imágenes se generan a partir de una representación mediante líneas de nivel de la topografía y demás objetos presentes en la escena, y el resultado final en cuanto a similitud con imágenes radar verdaderas es sumamente satisfactorio.

Se logró también optimizar el costo computacional del algoritmo mediante técnicas de clasificación geométrica, alcanzándose el requisito de revoluciones por minuto de un radar real, independientemente de la complejidad del escenario.

El modelo propuesto permitió incorporar fácilmente otros fenómenos secundarios como ruido de mar (sea clutter), pérdida de sintonía, efectos atmosféricos, o incluso reflexiones en estructuras del propio barco. Se implementó además toda la funcionalidad de un radar moderno, desde filtros como anti-sea clutter o anti-rain clutter, hasta el simple cambio de rango. También se implementaron las funciones ARPA que permiten capturar y seguir blancos, estimando tiempos y distancias mínimas de acercamiento.

Finalmente se incorporó este modelo como parte del sistema MELIPAL-R generando así una interesante herramienta para entrenamiento, la cual obtuvo recientemente la certificación de la OMI (Organización Marítima Internacional) que permite otorgar licencia internacional a los cursantes.

6 REFERENCIAS

- [1] http://www.transas.com/simulators/bridge_simulators/arpa/index.asp
- [2] Ch. Cook and M. Bernfeld, “Radar Signals: An Introduction to Theory and Applications”, Artech House Incorporated, ISBN 0890067333, (1994).
- [3] H. Meikle, “Modern Radar System” Artech House Incorporated, ISBN 1580532942. (2001)
- [4] Van Wyck, “The Radar Book”, ISBN 0870333267 (1984)
- [5] E. Dari, M. Vénere, “Algoritmos eficientes para la búsqueda del elemento de una red que contiene un punto dado”. *Mecánica Computacional*, Vol.10, 455-464, (1990)