

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
ESCUELA DE GRADUADOS
MAESTRÍA EN INFORMÁTICA



**“SISTEMA DE CONTROL DE TIRO
PARA TORPEDOS A BORDO DE SUBMARINOS”**

**TESIS PARA OPTAR EL GRADO ACADÉMICO DE
MAGÍSTER EN INFORMÁTICA**

Presentado por:

Alfredo CARPIO Ramírez

Lima – Perú

2006

RESUMEN EJECUTIVO

Desde sus inicios, la Guerra Submarina ha probado diferentes métodos y maneras para realizar sus procedimientos de búsqueda, aproximación y ataque a los buques blancos. La gran restricción de utilizar los sensores en forma pasiva, logró el desarrollo de la “Táctica Submarina”, la cual incluye procedimientos con diversas fórmulas matemáticas y algoritmos para llegar a la solución del problema de tiro.

El presente proyecto pretende incorporar a la nueva tecnología informática, parte de los algoritmos matemáticos probados a través de décadas, los mismos que son desarrollados manualmente utilizando pizarras acrílicas, ábacos de material plástico y muchos cálculos mentales. El nuevo sistema proporcionará soluciones en forma rápida y exacta, sirviendo de excelente soporte a las decisiones del comando durante las fases de aproximación y ataque submarino.

*A las dotaciones de los Submarinos de la Marina de Guerra del Perú,
A Dios por esta nueva oportunidad de superación*

*Nuestro sincero agradecimiento a todos los profesores catedráticos de la
Escuela de Graduados, en especial al Doctor Maynard Kong, por su
apoyo y comprensión al brindarnos muchas facilidades*

ÍNDICE

1.	Introducción	11
1.1.	Fases de la aproximación y ataque submarina	11
1.1.1.	Fase de contacto.....	12
1.1.2.	Fase de clasificación.....	12
1.1.3.	Fase de aproximación	13
1.1.4.	Fase de ataque.....	13
1.1.5.	Fase de retirada.....	13
2.	Marco Teórico.	14
2.1.	Terminología del control de tiro	14
2.2.	Términos básicos relacionados con el buque blanco	15
2.3.	Términos básicos relacionados con las armas	15
2.4.	Definiciones de términos básicos relacionados con el buque propio.....	17
2.4.1.	Rumbo propio (Rp).....	17
2.4.2.	Angulo de Conducción (AC).....	17
2.4.3.	Rumbo de Aproximación (RP).....	17
2.4.4.	Rumbo de Acercamiento (RN).....	17
2.4.5.	Rumbo de Retirada (RR).....	17
2.4.6.	Rumbo de Alejamiento (RL).....	18
2.5.	Definiciones de términos básicos relacionados con el buque blanco	18
2.5.1.	Rumbo del blanco (Rb)	18
2.5.2.	Distancia Actual (DA).....	18
2.5.3.	Distancia a la Derrota (DD).....	18
2.5.4.	Angulo de Encuentro (AE).....	18
2.5.5.	Angulo en la Proa (AP)	19
2.5.6.	Marcación Verdadera (Mv).....	19

2.5.7.	Corrida del Blanco (H)	19
2.6.	Definiciones de Términos básicos relacionados con las armas	19
2.6.1.	Angulo de Impacto (AI)	19
2.6.2.	Distancia de Adquisición (Da)	19
2.6.3.	Corrida Supuesta del Torpedo (CST)	20
2.6.4.	Angulo de Adelanto o de Deflexión (AD)	20
2.6.5.	Angulo de Giro (AG).....	20
2.6.6.	Radio de Giro del torpedo (Z)	20
2.6.7.	Avance Recto (M)	20
2.6.8.	Línea Base de TLT (P)	21
2.6.9.	Avance del torpedo (J).....	21
2.7.	Diagramas en la línea de mira.....	21
2.7.1.	Cálculo ALOSp y ALOSb.....	24
2.7.2.	Cálculo ALOS R.....	24
2.7.3.	Buque propio aproa al Blanco	24
2.7.4.	Ley de variación de marcación cero	24
2.7.5.	Ley de variación de marcación distinta a cero	25
2.7.6.	Consideraciones con respecto a la Ley de Variación de Marcación	25
2.8.	Las maniobras propias con relación al análisis de la línea de mira	26
2.8.1.	Maniobra Inicial	26
2.8.2.	Maniobras subsiguientes	27
2.9.	Análisis de la línea de mira en el disparo de torpedos de corrida recta.	28
2.10.	Conceptos básicos	28
2.10.1.	Distancia a la Derrota (DD)	28
2.10.2.	Obtención del Rb en función del AP	28
2.11.	Ábacos de ley de variación de marcación.....	29
2.11.1.	Escala “A”	29

2.11.2.	Escala “B”	29
2.11.3.	Escala “C”	29
2.11.4.	Escala “D”	29
2.12.	Ploteo tiempo marcación.....	30
2.12.1.	Situación 1.- Buque aproando al blanco (o con $V_p = 0$).....	31
2.12.2.	Situación 2.- Buque no aproa al Blanco:	32
2.12.3.	Situación 3.- Empleando dos piernas sucesivas:.....	32
2.12.4.	Situación 4.- Buque ficticio:	32
2.13.	Calculo de la distancia al blanco cuando el buque propio no aproa a este.....	32
2.13.1.	Procedimiento	33
2.14.	Relación que existe entre: ley de variación de marcación, alos r, y distancia.....	33
2.15.	Interpretación de curvas con respecto a ZZ del blanco curvas cóncavas (alejamiento)	34
2.16.	Interpretación de curvas con respecto a zz del blanco. Curvas conexas (acercamiento)	34
2.17.	Método Ekelund para determinación de distancia	35
2.17.1.	Características y limitaciones del Método Ekelund.....	35
2.18.	Ploteo Lynch	36
2.18.1.	Procedimiento	37
2.19.	Método COFFEY	38
2.19.1.	Procedimiento	38
2.19.2.	Factores limitativos	39
2.20.	Ploteo tiempo distancia	40
2.20.1.	Procedimiento	40
3.	Análisis y determinación de requisitos del sistema.	42
3.1.	Definición del Problema	42
3.2.	Justificación del sistema.....	42

3.3.	Objetivos	44
3.3.1.	Objetivos cualitativos para el Producto.....	44
3.3.2.	Objetivos cuantitativos para el Producto.....	44
3.3.3.	Objetivos Cualitativos para el Proceso.....	44
3.3.4.	Objetivos Cuantitativos para el Proceso.....	44
3.4.	Estrategias	44
3.5.	Instrumentos y técnicas a usar.	45
3.6.	Alcance.....	45
3.7.	Especificaciones de requisito del sistema	45
3.7.1.	Requerimiento Cuantitativo del Sistema.....	45
3.7.2.	Requerimiento Cualitativo del Sistema.....	46
3.7.3.	Requerimiento de hardware.....	46
3.7.4.	Requerimiento de software.....	46
3.7.5.	Requerimientos de Publicaciones.....	46
3.7.6.	Criterio de aceptación.....	46
4.	Diseño del sistema.	48
4.1.	Subsistemas componentes.....	48
4.1.1.	Hardware.....	48
4.1.2.	Software.....	48
4.2.	Subsistema de interfaz con el usuario	48
4.2.1.	Entorno visual con el usuario	48
4.2.2.	Entrada y validación de datos.....	49
4.3.	Subsistema de procesamiento	49
4.3.1.	Módulo de Ingreso.....	49
4.3.2.	Módulo de Proceso de Distancia.	50
4.3.3.	Módulo de Proceso de Diagrama en la Línea de Mira (LOS)....	50
4.3.4.	Módulo de Proceso de "Ploteo Tiempo Marcación" (TBP).....	50
4.3.5.	Módulo de proceso de "Ploteo Geográfico"	50
4.3.6.	Módulo de Salida de "Instructor de Ataque"	51

4.3.7. Módulo de Salida General	51
4.4. Características del usuario.....	51
4.5. Diccionario de términos	51
4.6. Especificación del Plan de Pruebas.....	53
5. Codificación.	57
6. Conclusiones y recomendaciones	91
6.1. Conclusiones	91
6.2. Recomendaciones.....	92

ANEXOS

Anexo 01: Estimación de costos.	94
---------------------------------	----

ÍNDICE DE CUADROS

Cuadro 2.1: Interpretación curvas cóncavas en alejamiento	34
Cuadro 2.2: Interpretación curvas conexas en acercamiento	35

ÍNDICE DE GRÁFICOS

Gráfico 2.1: Definiciones y símbolos usados en el problema de control de tiro	16
Gráfico 2.2: Ley de variación de marcación	22
Gráfico 2.3: Diagrama en la línea de mira	23
Gráfico 4.1: Entorno visual con el usuario	49
Gráfico 4.2: Entrada y validación de datos Ejemplo N° 1.	54

Gráfico 4.3: Solución Ejemplo N° 1.	54
Gráfico 4.4: Entrada y validación de datos Ejemplo N° 2.	55
Gráfico 4.5: Solución Ejemplo N° 2.	55
Gráfico 4.6: Entrada y validación de datos Ejemplo N° 3.	56
Gráfico 4.2: Solución Ejemplo N° 3.	56

ÍNDICE DE FOTOS

Foto N° 2.1: Ábaco de Ley de Variación versión norteamericana	30
Foto N° 2.2: Ábaco de Ley de Variación versión alemana	30
Foto N° 3.1: Torpedo Data Computer (TDC) de tecnología obsoleta.	43

CAPITULO I

1. Introducción

La historia de los sumergibles y submarinos es centenaria, pues desde el Siglo XVI el hombre comenzó a pensar seriamente en diseñar uno de estos navíos. En 1578, William Bourne, un artillero retirado de la Marina Real Británica, diseñó la primera embarcación para la navegación submarina. Era un barco completamente cerrado que podía ser sumergido y conducido con remos bajo la superficie del agua. Con un marco de madera y piel impermeable, estaba basado en tanques de lastre que eran llenados para sumergirse y vaciados para emerger, los mismos principios que hoy se usan en los submarinos.

Durante la Guerra Civil de los Estados Unidos, los bandos en conflicto probaron diversos prototipos de sumergibles como el Caimán y la Ballena, pero todos fracasaron. El Hunley, rediseñado a partir de un modelo de Horace Lawson Hunley, inventor que se ahogó al probar su creación, logró atacar y hundir al barco de vapor Housatonic en la bahía de Charleston, tras lo cual desapareció. Este fue el primer ataque submarino exitoso registrado en la historia, e inició la carrera por descubrir los mejores procedimientos de aproximación y ataque.

1.1. Fases de la aproximación y ataque submarina

La aproximación y Ataque Submarina contemplaba tradicionalmente cuatro (4) fases bien definidas, las cuales eran: Fase de Contacto, Aproximación, Ataque y Retirada. Actualmente

estas cuatro fases se han visto incrementadas con una adicional, la cual corresponde a la Fase de Clasificación. Esta fase se debe fundamentalmente a que los sensores modernos permiten la detección de blancos a gran distancia, ésta detección lejana obliga a planificar correctamente el empleo de sensores, asimismo origina que se presente una dificultad mayor en el proceso de clasificar e identificar correctamente al contacto detectado.

1.1.1. Fase de contacto

La fase de contacto se inicia en el momento de la detección de un blanco. La detección de blancos requiere de una vigilancia metódica, continua y a largo plazo de uno, o varios campos de energía y de la capacidad para discernir cualquier anomalía que se produzca dentro de ese (esos) campo (s). El objetivo fundamental de la Fase de Contacto es el de establecer la dirección de movimiento del blanco.

1.1.2. Fase de clasificación

La fase de clasificación se inicia también con la detección inicial y se mantiene presente hasta que se determine la identificación del blanco, o hasta el momento en que se efectúa el lanzamiento del arma, lo que ocurra primero.

La fase de clasificación consiste básicamente de dos etapas: la primera de ellas corresponde a la Clasificación propiamente dicha de la anomalía detectada y la segunda a la Identificación de la plataforma que la produce.

Etapas de Clasificación

La etapa de clasificación comprende el análisis de la señal para determinar el tipo de fuente que la produce

Etapa de identificación

La identificación corresponde en distinguir por medio del análisis de los caracteres del contacto detectado, de quien se trata. Esta etapa es quizás la más difícil de lograr, pues en la mayoría de los casos el proceso de identificación se logrará empleando el periscopio como única alternativa.

1.1.3. Fase de aproximación

El comandante normalmente conduce a una aproximación con los siguientes objetivos en mente:

- Ubicarse dentro del alcance efectivo del arma
- Permanecer sin ser detectado, al menos, hasta después de haber lanzado las armas con éxito.
- Determinar los parámetros del blanco con suficiente exactitud para lograr que el arma sea colocada dentro de sus distancias de adquisición al blanco.

Para lograr estos objetivos, debe adoptar diversos rumbos y velocidades con el propósito de obtener soluciones en los diferentes ploteos que realiza su Team de Ataque.

1.1.4. Fase de ataque

Es el momento en que el submarino realiza el lanzamiento de torpedos, luego que logra una posición ideal y su sistema de control de tiro indican solución.

1.1.5. Fase de retirada

Son los diferentes medios de evasión que realiza el submarino luego de concluir su ataque.

CAPITULO II

2. Marco Teórico.

El problema de control de tiro en los submarinos tiene cuatro características relevantes:

- Traquear un blanco por medios pasivos empleando el sonar, periscopio y/o cualquier otro sensor que se disponga a bordo.
- Determinar la distancia, rumbo y velocidad del blanco empleando técnicas pasivas de análisis.
- Ubicarse dentro del alcance efectivo de las armas.
- Colocar el arma dentro de su distancia de adquisición al blanco.

Un conocimiento apropiado de la geometría del control de tiro, es decir de la relación existente entre el buque propio, las armas y el blanco permite simplificar estos aspectos.

2.1. Terminología del control de tiro¹

- Rumbo propio (Rp)
- Angulo de Conducción (AC)
- Rumbo de aproximación (RP)
- Rumbo de Acercamiento (RN)
- Rumbo de Retirada (RR)
- Rumbo de Alejamiento (RD)

¹ Los triángulos que se presentan en el gráfico 2.1, permitirán comprender mejor estos términos.

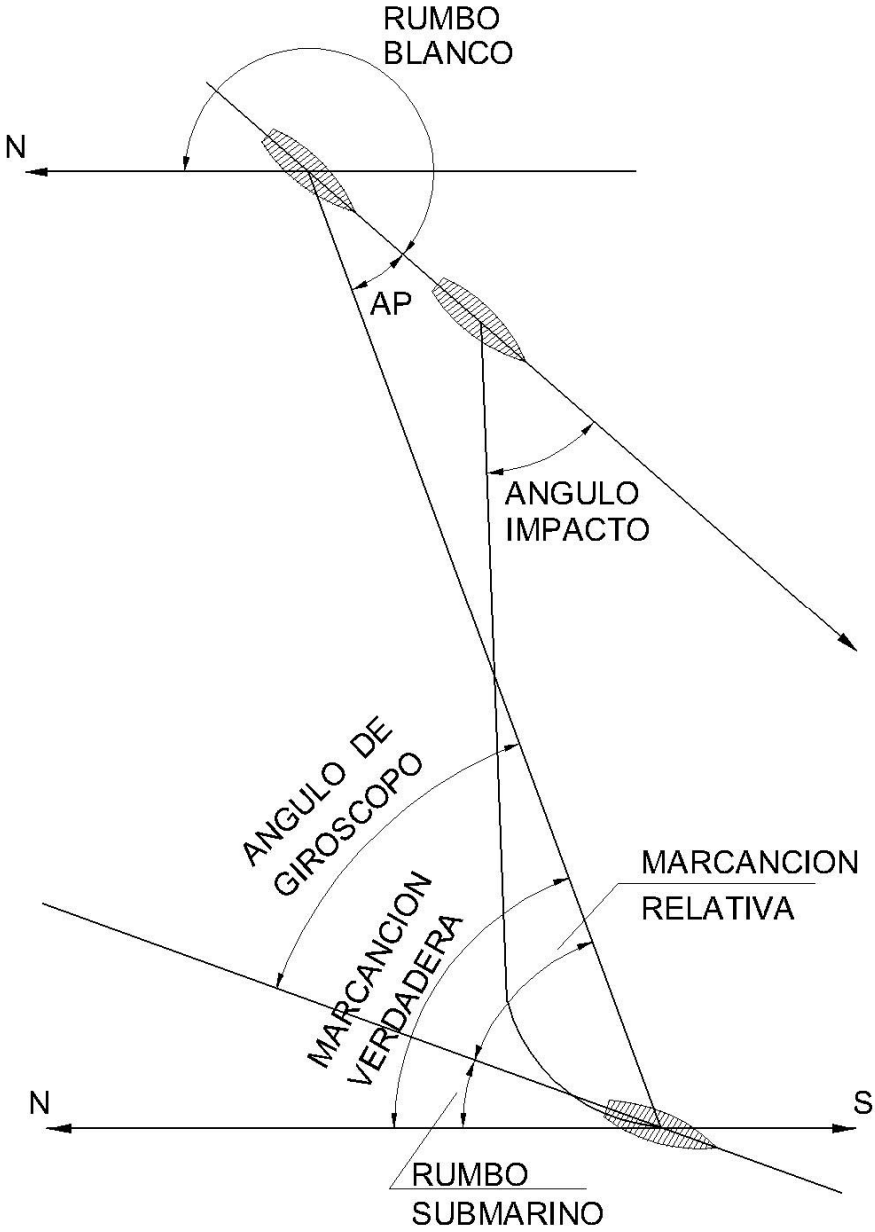
2.2. Términos básicos relacionados con el buque blanco

- Rumbo del blanco (Rb)
- Distancia Actual (DA)
- Distancia a la Derrota (DD)
- Angulo de Encuentro (AE)
- Angulo en la Proa (AP)
- Marcación Verdadera (Mv)
- Corrida del blanco (H)

2.3. Términos básicos relacionados con las armas

- Angulo de Impacto (AI)
- Distancia de Adquisición (Da)
- Corrida Supuesta Torpedo (Cst)
- Angulo de Adelanto (AD)
- Angulo de Giro (AG)
- Radio de Giro (Z)
- Avance Recto (M)
- Línea Base de TLT (P)
- Avance del Torpedo (J)

Gráfico 2.1: Definiciones y símbolos usados en el problema de control de tiro



2.4. Definiciones de términos básicos relacionados con el buque propio

2.4.1. Rumbo propio (Rp)

Es el rumbo al cual navega el buque propio

2.4.2. Angulo de Conducción (AC)

Es el ángulo formado entre la marcación verdadera al blanco y el rumbo propio.

2.4.3. Rumbo de Aproximación (RP)

Es el rumbo del buque propio a ángulo recto con la marcación al blanco y en sentido de acercarse a su derrota. Se emplea durante la aproximación para llegar a una posición favorable para el lanzamiento cuando el ángulo en la proa es grande.

2.4.4. Rumbo de Acercamiento (RN)

Es el rumbo del buque propio perpendicular a la derrota del blanco en el sentido de acercarse a éste. Se emplea para acercarse a la derrota al blanco en el mínimo de tiempo posible.

2.4.5. Rumbo de Retirada (RR)

Es el rumbo del buque propio a ángulo recto con la marcación al blanco y en sentido de alejarse a su derrota. Corresponde a la inversa del rumbo de aproximación. Usualmente el submarino toma este rumbo con respecto a la última marcación de fuego después de lanzar el último

torpedo no filoguiado.

2.4.6. **Rumbo de Alejamiento (RL)**

Es el rumbo del buque propio perpendicular a la derrota del blanco en el sentido de alejarse a éste. Corresponde a la inversa del rumbo de Acercamiento, se emplea para alejarse de la derrota en el menor tiempo posible.

2.5. **Definiciones de términos básicos relacionados con el buque blanco**

2.5.1. **Rumbo del blanco (Rb)**

Es el rumbo al cual navega el buque blanco

2.5.2. **Distancia Actual (DA)**

Es la distancia al blanco en un momento determinado, en el momento del lanzamiento se convierte en Distancia de Fuego (DF)

2.5.3. **Distancia a la Derrota (DD)**

Es la distancia en yardas más corta entre el buque propio y la derrota del blanco.

2.5.4. **Angulo de Encuentro (AE)**

Es el ángulo de interceptación formado entre el rumbo del blanco y el rumbo inverso del submarino, medido desde la proa del blanco de 0° a 180° por babor o estribor del mismo.

2.5.5. **Angulo en la Proa (AP)**

Es el ángulo comprendido entre el rumbo del blanco y la recíproca de la marcación verdadera al blanco. Se mide de 0° a 180° por babor o estribor.

2.5.6. **Marcación Verdadera (Mv)**

Es el ángulo comprendido entre el norte verdadero y la marcación al blanco. Se mide de 0° a 360° y se expresa por medio de tres dígitos.

2.5.7. **Corrida del Blanco (H)**

Es la distancia en yardas recorridas por el blanco desde el momento de inicio de la corrida del torpedo.

2.6. **Definiciones de Términos básicos relacionados con las armas**

2.6.1. **Angulo de Impacto (AI)**

Es el ángulo formado por la interceptación de la derrota del blanco con la corrida del torpedo, medido desde la proa del blanco de 0° a 180° por babor o estribor.

2.6.2. **Distancia de Adquisición (Da)**

Distancia de adquisición o Punto laminar, es un valor teórico medido en yardas que se introduce al SCT y que como resultado ubica al torpedo en una posición final aproando al blanco y a una distancia de éste igual al valor de D_a graduado. Teóricamente este valor es seleccionado teniendo en cuenta tres factores: velocidad del blanco, probabilidad de detección del torpedo y probabilidad de impacto. Para cada valor de D_a corresponde un nuevo valor

de rumbo del torpedo; si se introdujera $D_a = \text{cero}$ yardas el torpedo seguiría a rumbo de colisión con el blanco, asimismo debe considerarse como máximo valor de distancia de adquisición el máximo alcance predicho de sonar para el torpedo.

2.6.3. **Corrida Supuesta del Torpedo (CST)**

Es la distancia en yardas medida desde el periscopio al punto de interceptación o de impacto del torpedo con la derrota del blanco

2.6.4. **Angulo de Adelanto o de Deflexión (AD)**

Es el ángulo comprendido entre la marcación verdadera al blanco y la corrida supuesta del torpedo en el momento de fuego.

2.6.5. **Angulo de Giro (AG)**

Es el ángulo medido entre el eje longitudinal del TLT y el rumbo del torpedo. Se mide de 0° a 180° por babor o estribor.

2.6.6. **Radio de Giro del torpedo (Z)**

Es el radio de la trayectoria circular del torpedo, desde el momento en que termina el avance recto hacia el comienzo del rumbo final, se mide en yardas

2.6.7. **Avance Recto (M)**

Es el recorrido recto inicial del torpedo medido en yardas, desde la salida del TLT hasta el inicio del giro dado por el ángulo de giroscopio

2.6.8. **Línea Base de TLT (P)**

Es la distancia, medida en yardas, desde el punto de los periscopios hasta los tapabocas de los TLT

2.6.9. **Avance del torpedo (J)**

Es la distancia en yardas medida en forma perpendicular entre la corrida final del torpedo y la línea imaginaria que corre desde el tapaboca del TLT en forma paralela a la corrida final del torpedo.

2.7. **Diagramas en la línea de mira**

La Línea de Mira (LOS por su sigla en inglés) es una línea que parte de un punto de referencia del buque propio y llega a un punto de blanco. En un diagrama LOS el blanco y el buque propio son representados normalmente por vectores, la longitud de los mismos denota el valor de la velocidad y sus direcciones denotan el rumbo.

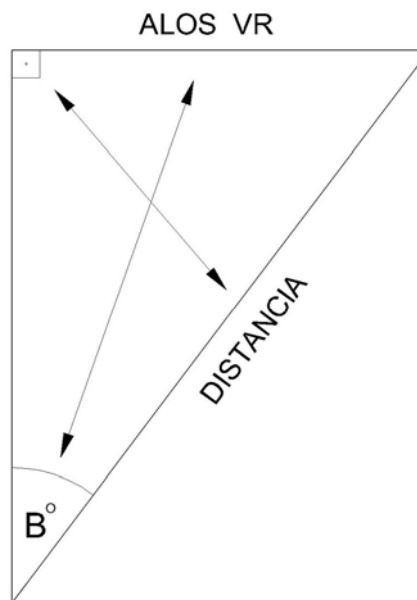
Todos los vectores de velocidad en un diagrama LOS pueden ser reducidos a componentes “a través” y en “la línea de mira “ (ALOS e ILOS respectivamente). El trabajar con vectores permite realizar operaciones algebraicas de suma las cuales ayudarán a definir el vector resultante luego de una descomposición de vectores.

Es conveniente representar en el análisis de la línea de mira, la dirección del vector ALOS (propio y del blanco) mediante las letras D (derecha) o I (izquierda) de manera de poder ejecutar la suma algebraica de forma más sencilla.

Las componentes resultantes de ALOSr e ILOSr son asociados con la “ley de variación de marcación B° ” y con la “razón de distancia R” respectivamente.

La Ley de variación de marcación corresponde al régimen de cambio de marcación verdadera del blanco expresado en grados/minuto medido hacia la derecha o izquierda y en la misma dirección de los ALOSr.

Gráfico 2.2: Ley de variación de marcación



La razón de distancia corresponde al valor de ILOSr pero expresado en yardas por minuto.

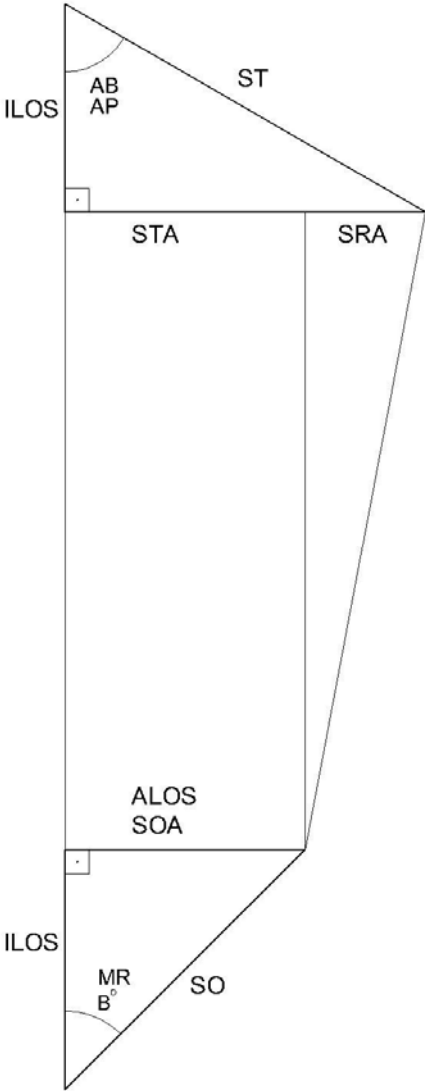
Al resolver la velocidad a través de la línea de mira partiendo de los respectivos vectores de velocidad y de su ángulos asociados llegamos a las siguientes fórmulas de interés:

$$\text{ALOS } p = \text{Sen } AC$$

$$\text{ILOS } p = V_p \text{ Sen } (90-AC) \text{ ó } VP \text{ cos } AC$$

Los diagramas que se presentan a continuación permitirán comprender mejor estas relaciones.

Gráfico 2.3: Diagrama en la línea de mira



2.7.1. Cálculo ALOSp y ALOSb

$$\text{ALOSp} = V_p \text{ Sen } M_r$$

$$\text{ALOSb} = V_b \text{ Sen } A_P$$

2.7.2. Cálculo ALOS R

Rp en sentido del movimiento del blanco

$$\text{ALOS R} = \text{ALOS b} - \text{ALOSp}$$

Rp en sentido contrario al Mov. Del blanco

$$\text{ALOS R} = \text{ALOSb} + \text{ALOSp}$$

2.7.3. Buque propio aproa al Blanco

Cuando el buque propio aproa al blanco se anula la componente propia de velocidad a través de la línea de mira, siendo de esta manera factible determinar rápidamente el posible rumbo del blanco (sentido de movimiento del mismo). Normalmente se adopta también mínima velocidad (3 nudos), de manera de poder mantener estable el rumbo y la profundidad.

Para $\text{ALOSp} = 0$

$$\text{ALOS b} = \frac{B^\circ \times DA}{1936}$$

2.7.4. Ley de variación de marcación cero

Cuando la ley de variación de marcación es cero, significa que los ALOSp y los ALOSb son iguales, de manera que los ALOS R son cero; siendo de esta manera posible determinar el rumbo del blanco en función de la velocidad del blanco y de la velocidad propia a través de la línea de

mira. Es decir que conociendo solamente una variable (V_b) de las tres por determinar (R_b , V_b y distancia) es posible calcular las otras dos.

Para $B^\circ = 0$

$$\text{Sen AP} = \frac{\text{ALOS } p}{V_b}$$

2.7.5. Ley de variación de marcación distinta a cero

Cuando la ley de variación de marcación no es cero, se cumple con:

“La ley de variación de marcación es directamente proporcional a la velocidad resultante a través de la línea de mira, e inversamente proporcional a la distancia”

$$B^\circ = \frac{\text{ALOS } R \times 1936}{DA}$$

2.7.6. Consideraciones con respecto a la Ley de Variación de Marcación

Una gran Ley de Variación de Marcación puede ser por un lado una indicación de que el blanco se encuentre relativamente cerca, y por otro lado que tenga una velocidad a través de la línea de mira grande. Con esto quiere decirse que el valor de la Ley de variación de Marcación, por sí sola, no es indicio de la distancia al blanco, ni del rumbo del mismo, por lo que es indispensable incluir otras observaciones como por ejemplo: un incremento en el ruido que escucha el Sonarista determina que el blanco se acerca, si el incremento es rápido significará que el AP es pequeño; si las revoluciones son rápidas hacen deducir altas velocidades; si el ruido no es

fuerte y las condiciones de propagación son normales, entonces el blanco estará cerca.

Asimismo, las maniobras que efectúa el buque propio deben tender a controlar la Ley de Variación de Marcación, no es lo mismo por ejemplo una pierna en sentido contrario al movimiento del blanco que en el mismo sentido, en el primer caso existirá una mayor Ley de Variación de Marcación que en el segundo de ellos.

Un adecuado control de la ley de Variación de Marcación es condición indispensable para no solamente obtener una adecuada solución de los parámetros del blanco en los distintos ploteos, sino para llegar a ocupar una posición favorable para el lanzamiento de las armas.

Ud. Encontrará en los ejercicios propuestos un grupo de ellos para el cálculo de distancia al blanco, en general es sencillo efectuar el cálculo de distancia, lo importante es ser ordenado y mantener una secuencia de las variables que se desean obtener sobre la base de los datos disponibles.

2.8. Las maniobras propias con relación al análisis de la línea de mira

2.8.1. Maniobra Inicial

Normalmente un Submarino durante la Fase de Contacto aproa al blanco y disminuye su velocidad con la finalidad de eliminar la componente propia de velocidad a través de la línea de mira y poder de esta manera analizar el comportamiento del blanco, es decir determinar si éste se acerca o se aleja y por qué banda lo hace.

2.8.2. Maniobras subsiguientes

Una vez concluida la fase de contacto, el Oficial atacante debe decidir su siguiente maniobra pudiendo ser, de manera general, una de las dos siguientes:

Un rumbo de conducción en sentido del movimiento del blanco

Este tipo de maniobra tiene las siguientes ventajas: Permite acercarse a la derrota del blanco, disminuye la distancia al PMA y a la vez aumenta el tiempo para que esto ocurra. Como desventaja presenta una ley de variación de marcación pequeña, lo que dificultará el trabajo de los ploteos.

Un rumbo de conducción en sentido contrario al movimiento del blanco

Este tipo de maniobra tiene la ventaja de general una gran ley de variación de marcación indispensable para el buen trabajo de los ploteos (a pesar de que el blanco se encuentre relativamente lejano). Como desventaja presenta una disminución del tiempo para llegar el PMA y un aumento de la distancia al mismo. Si el Oficial atacante no controla la cantidad de grados en marcación que gana el blanco puede llegar a perderlo.

Como situación especial podemos mencionar el caso en que al efectuar una maniobra, sea esta la inicial o la descrita anteriormente, podamos obtener ley de variación de marcación cero. Esta situación puede ser generada

intencionalmente o de forma casual. Lo cierto es que la ventaja de esta situación es grande para obtener la solución del blanco ya que al ser los ALOS propios iguales a los ALOS del blanco, los ALOS R son CERO, de esta forma se obtiene:

$$\text{Sen AP} = \frac{\text{ALOSp}}{\text{Vb}}$$

2.9. Análisis de la línea de mira en el disparo de torpedos de corrida recta.

Un lanzamiento de torpedos de corrida recta involucra una solución del valor de ángulo de deflexión (AD) tal, que general un valor de ALOSt igual al valor ALOSb, haciendo que la ley de variación de marcación relativa entre el torpedo y el blanco sea CERO. Esta condición debe mantenerse durante toda la corrida del torpedo para que llegue a impactar al blanco².

2.10. Conceptos básicos

2.10.1. Distancia a la Derrota (DD)

$$\text{DD} = \text{DA} \times \text{Sen AP}$$

2.10.2. Obtención del Rb en función del AP

Rb = Inversa marcación verdadera - AP (para AP por estribor)

Rb = Inversa marcación verdadera + AP (para AP por babor)

² El torpedo recibe el valor de Angulo de Giro y no de Angulo de Deflexión, por lo que éste debe ser calculado

2.11. Ábacos de ley de variación de marcación

Uno de los propósitos de este tema de tesis, es el de reemplazar estas herramientas de cálculo manuales por un software que realice los mismos cálculos. Estos son unos ábacos circulares de material plástico, que permiten calcular distancias, velocidades y/o ángulos por medio de la relación de cuatro (4) escalas de base logarítmica; de afuera hacia adentro estas escalas son:

2.11.1. Escala “A”

Escala de velocidad graduada de 0.6 a 50 y que dependiendo del cálculo a efectuarse representa:

- Velocidad en nudos
- Velocidad a través de la línea de mira en nudos

2.11.2. Escala “B”

Escala de ley de variación de marcación graduada de 0.25 a 20

2.11.3. Escala “C”

Escala de distancias, graduada de 500 yardas a 45 Kyds.

2.11.4. Escala “D”

Escala logarítmica de base senos, en la que se representa ángulos de 1° a 90°

Foto N° 2.1: Ábaco de Ley de Variación versión norteamericana

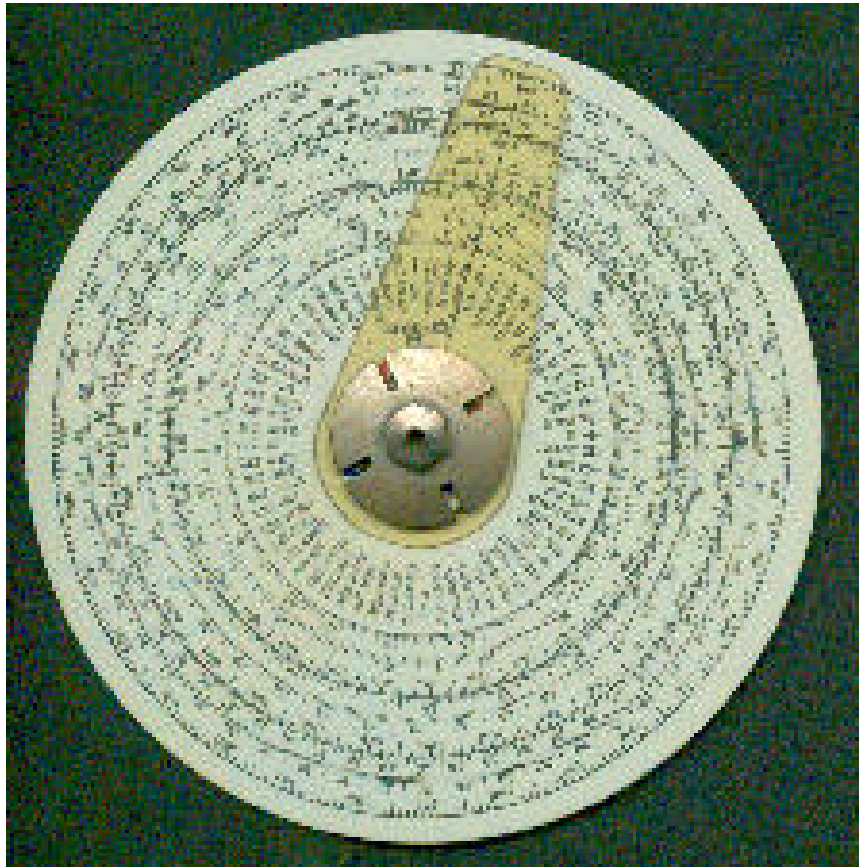
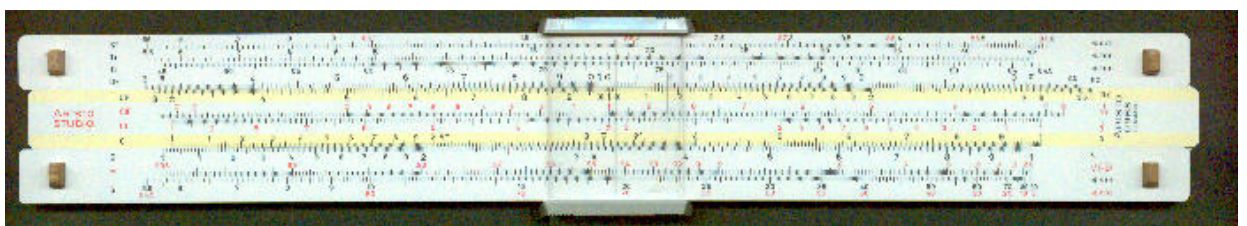


Foto N° 2.2: Ábaco de Ley de Variación versión alemana



2.12.Ploteo tiempo marcación

El ploteo Tiempo - Marcación es conocido como Ploteo Bernard o Ploteo TBP, se basa exclusivamente en las marcaciones de sonar que se obtienen de un blanco determinado. Se realiza sobre un plano de coordenadas cartesianas ubicando las marcaciones en el

eje de las abscisas (eje x) y el tiempo en el eje de las ordenadas (eje Y).

Las marcaciones sucesivas que se obtengan del blanco ploteadas versus el tiempo respectivo en que obtuvieron generarán puntos en este plano, que unidos convenientemente representarán una curva (o una recta), indicándonos de esta manera el movimiento relativo del blanco.

La tangente, o dicho de otra manera la pendiente de la curva, en cualquier punto que se mida nos indicará el valor de la Ley de variación de marcación (B°) respectiva. El trazado adecuado de la curva nos permitirá compensar o filtrar, cualquier dispersión existente entre las marcaciones ploteadas.

La escala a considerar será:

1. Para las marcaciones: un (1) grado o dos (2) por cuadrado
2. Para el tiempo: una sola escala de un (1) minuto por cuadrado

Con este ploteo y en función a las piernas que se realice es posible obtener lo siguiente:

2.12.1. Situación 1.- Buque aproando al blanco (o con $V_p = 0$)

- Determinación del movimiento relativo del blanco, identificando si el blanco se aleja o se acerca y por que banda lo hace
- Estimación del Angulo de la proa (AP) del blanco, en función a la inclinación de la curva obtenida
- En base al AP estimado y la velocidad supuesta, o estimada por el Comandante, se puede calcular la distancia inicial al minuto “cero”

2.12.2. Situación 2.- Buque no aproa al Blanco:

- Se podrá calcular la distancia al blanco en cualquier minuto empleando los valores de rumbo del blanco (R_b) y velocidad del blanco (V_b) dados por el Comandante.

2.12.3. Situación 3.- Empleando dos piernas sucesivas:

- Se podrá calcular la distancia al blanco para el minuto correspondiente al corte (punto “S”) de las curvas Tiempo Marcación generadas por el Buque propio al haber cambiado éste su rumbo y velocidad
- Se podrá calcular la distancia al blanco por el método de la Cotangente de la Ley de Variación de Marcación

2.12.4. Situación 4.- Buque ficticio:

- La Proyección de la curva obtenida permitirá determinar marcaciones y valores de ley de variación de marcación para el buque ficticio, anteriores o posteriores a la data disponible en un momento dado, para su aplicación posterior en otros ploteos, por ejemplo para el Ploteo Speiss

2.13. Calculo de la distancia al blanco cuando el buque propio no aproa a este.

En el ploteo TBP se podrá calcular la distancia al blanco en cualquier minuto de interés empleando los siguientes valores:

- Rumbo del blanco obtenido del mismo ploteo o proporcionado por el atacante
- Velocidad del blanco proporcionada por el atacante.

2.13.1. Procedimiento

- Trace la línea de mira
- Determine y grafique el vector propio
- Calcule ALOSp
- Determine AP y grafique vector del blanco
- Calcule ALOSb
- Determine ALOS R
- Mida B° para el minuto de interés
- Calcule distancia

2.14. Relación que existe entre: ley de variación de marcación, a los r, y distancia

Para interpretar una curva del ploteo Tiempo Marcación debe entenderse claramente la relación que existe entre: ley de variación de marcación, ALOS R, y distancia

$$B^{\circ} = \frac{ALOS R \times 1936}{DA}$$

Esta fórmula expresa que la ley de variación de marcación es directamente proporcional a los ALOS R, siendo éstos últimos a su vez función del ángulo en la proa, velocidad del blanco y del vector propio de rumbo y velocidad. Si tanto el buque propio como el blanco no maniobraran y la velocidad relativa fuera mayor de CERO nudos, entonces los ALOS R se incrementarán conforme aumente el valor del AP. Esto continuará hasta llegar a un valor máximo, el cual ocurrirá en el PMA ($AP = 90^{\circ}$) a partir del cual comenzará a disminuir. Del mismo modo la B° continuará incrementándose conforme el blanco se acerque al PMA, para decrecer posteriormente.

Cualquier maniobra producida por el buque propio o el blanco variará los ALOS R, y por consiguiente se interrumpirá la continuidad de la curva existente, comenzando una nueva curva desde el punto de maniobra.

2.15. Interpretación de curvas con respecto a ZZ del blanco curvas cóncavas (alejamiento)

A continuación presentaremos reglas generales con respecto a las curvas que se aprecian en el ploteo TBP cuando el blanco se aleja. Aquí hay que recordar que cuando el AP es mayor de 90° el seno del ángulo en la proa corresponde al seno del suplemento del mismo. Esto significa que cuando el AP sea $> 90^\circ$ pero cercano a éste, el seno será mayor que el correspondiente a un ángulo en la proa cercano a 180° .

Cuadro 2.1: Interpretación curvas cóncavas en alejamiento

CASOS	Análisis con respecto Rb	Análisis con respecto Vb
$B^\circ II < B^\circ I$	Ang. Proa AUMENTA	Velocidad : DISMINUYE
$B^\circ II > B^\circ I$	Ang. Proa. DISMINUYE	Velocidad: AUMENTA

2.16. Interpretación de curvas con respecto a zz del blanco. Curvas conexas (acercamiento)

A continuación se presentan las reglas generales con respecto a las curvas que se aprecian en el ploteo TBP cuando el blanco se acerca.

Cuadro 2.2: Interpretación curvas conexas en acercamiento

CASOS	Análisis con respecto Rb	Análisis con respecto Vb
B° II < B° I	Ang. Proa DISMINUYE	Velocidad : DISMINUYE
B° II > B° I	Ang. Proa. AUMENTA	Velocidad: AUMENTA

2.17.Método Ekelund para determinación de distancia

El método Ekelund para determinación de distancia permite obtener el valor de la distancia al blanco empleando datos obtenidos por la maniobra propia sin necesidad de asumir valor alguno del blanco.

La fórmula que rige el cálculo de la distancia EKELUND es la siguiente:

$$D \text{ Ekelund} = \frac{\text{Dif ALOSp}}{\text{Dif B}^\circ} \times 1,936$$

El desarrollo de los cálculos es en sí sencillo, el problema se presenta para determinar el tiempo para el cual la distancia Ekelund hallada es válida.

2.17.1. Características y limitaciones del Método Ekelund

- Este método requiere que el buque propio realice dos piernas
- Como en todos los ploteos el método establece que el blanco no altere sus rumbo ni su velocidad durante el tiempo que demore la realización del mismo.
- Las piernas deben tener una duración mínima de tres (3) minutos y máxima de cinco (5) minutos.
- Maniobre y estabilice el Rp y la Vp tan rápido como sea posible.

- Minimice la diferencia que exista entre la M_v (1) y la M_v (2), es decir entre los puntos de medición de B° .

$ \Delta M_v =$	$M_v 1 - M_v 2$	$< 15\%$
-----------------	-----------------	----------

- Maniobre para que la diferencia de $ALOS_p$ sea mayor de 6 nudos

$\Delta ALOS_p > 6$ nudos

- Maximice la diferencia entre $B^\circ 1$ y $B^\circ 2$ para que sea mayor de $1^\circ/\text{min}$

$B^\circ 1 - B^\circ 2 > 1^\circ/\text{min}$
--

- Mida los parámetros lo más preciso posible

2.18. Ploteo Lynch

El Ploteo Lynch es un ploteo relativo que se basa en dos tipos de información para la obtención de los datos del blanco; una corresponde a la marcación verdadera proporcionada por el sonar, y la otra corresponde a la Ley de Variación de Marcación (B°) proveniente del ploteo Tiempo Marcación.

Este ploteo se puede desarrollar sobre una Rosa de Maniobra común pero se necesita además una escala apropiada de Ley de Variación de Marcación sobre esta rosa se trazan las marcaciones verdaderas al blanco y sobre cada marcación se plotea la correspondiente B° . La unión de los puntos de B° ploteados nos dará la Dirección de Movimiento Relativo (DMR), en base a este dato será posible calcular posteriormente los siguiente:

- Rumbo del blanco para una velocidad asumida del mismo
- B° en el PMA
- Distancia al PMA
- Distancia al Blanco

2.18.1. Procedimiento

Determinación del Rumbo del blanco

- Como se mencionó anteriormente al inicio del ploteo se trazan las marcaciones de sonar correspondientes a los minutos respectivos de una pierna. Sobre estas marcaciones se grafican los puntos correspondientes a B°, los cuales se obtienen del ploteo Tiempo Marcación; la unión de estos puntos nos dará la DMR, la cual se traslada al vector rumbo /velocidad propio, del punto de corte con el círculo de velocidad del blanco nos indicará el rumbo del mismo.

Determinación de la distancia al PMA

- Una vez obtenida la DMR del blanco se determina el punto correspondiente al PMA procediéndose a medir la B° correspondiente (B° PMA), este valor de B° nos va a servir para calcular la distancia al PMA. Luego se procederá a medir el valor del vector velocidad relativa (vector rm). Aquí hay que hacer notar que en el PMA el valor del vector velocidad relativa corresponde al valor del vector ALOS R, con estos dos valores (B° PMA y ALOS R) se procede a calcular la distancia PMA..

$$D \text{ PMA} = \frac{1936 \times \text{ALOS R}}{B^\circ \text{ PMA}}$$

- Este valor de distancia procedemos a graficarlo sobre la marcación al PMA, en este punto se traza una paralela a la DMR encontrado previamente de manera de obtener la línea de movimiento relativo del blanco.
- El corte de la LMR con cualquier marcación permitirá obtener directamente la distancia al blanco.

Determinación de los datos del blanco por efecto del cambio del Rp y Vp.

- Si por un cambio de rumbo y/o velocidad propia se obtiene una nueva curva en el TBP, es posible utilizar esta información para efectuar un nuevo ploteo Lynch sobre la misma rosa, con la finalidad de que la segunda DMR al ser trasladada al nuevo vector Er2 corte a la DMR1 trazada originalmente, este corte de DMR determina el rumbo y velocidad del blanco³.

2.19.Método COFFEY

El método Coffey es un ploteo que se efectúa sobre una rosa de maniobra, se basa en una técnica para determinar el Rumbo del blanco asumiendo que la ley de variación inicial es cero.

2.19.1. Procedimiento

- Plotee el vector rumbo y velocidad propio (er) sobre la rosa de maniobra

³ Por la exactitud del ploteo Lynch este debe ser considerado como un ploteo complementario, ya que la misma depende de tener un valor relativamente grande de Ley de variación de marcación, la cual no se logra sino en la etapa final de la fase de aproximación debido a los grandes alcances de detección que se logra en la actualidad.

- Plotee la marcación al contacto y su recíproca
- Asuma que la B° es cero, determine ahora el rumbo del blanco trasladando la marcación trazada al vector er (la marcación se convierte en DMR), de manera de encontrar con la velocidad estimada del blanco dos rumbos, uno de ellos corresponderá al rumbo de acercamiento y el otro al alejamiento
- Luego de haber realizado esta operación es necesario corregir el rumbo hallado, debido a que inicialmente asumimos que la ley de variación de marcación era cero, pero en realidad su valor absoluto era mayor que cero.
- Para encontrar la corrección multipliquemos el valor de la ley de variación de marcación correspondiente a la marcación trazada por el factor 50, aplique el resultado para los rumbos de acercamiento y/o alejamiento hallados anteriormente, la corrección se aplica en dirección opuesta del sentido de la ley de variación de marcación.

2.19.2. Factores limitativos

Este método presenta limitaciones para su aplicación, pero si se cumple con los siguientes requisitos fácilmente podremos hallar el resultado del rumbo del blanco con mucha exactitud.

- La ley de variación de marcación debe ser menor a $1^\circ/\text{min}$. Mientras la B° sea más cercana a cero, el resultado será más exacto.
- La exactitud del método también dependerá de que la velocidad del blanco se encuentre centrada en los 15 nudos y que el AP esté comprendido entre los 10° y 45°

- Si no se cumple con estos requisitos aplique otra técnica para solución de los datos del blanco, pues el resultado obtenido no será confiable.

2.20.Ploteo tiempo distancia

El ploteo Tiempo Distancia presenta un gráfico de las distancias calculadas por todas las técnicas de Análisis de Movimiento de Blancos descritas anteriormente; la aplicación de una línea que compense todas estas distancias ploteadas, permitirá obtener una estimación adecuada de la solución de distancia. Esta estimación de distancia puede ser usada para predecir el tiempo que le tomará al buque propio llegar a ocupar estación dentro del alcance efectivo de sus armas.

2.20.1. Procedimiento

- Este ploteo se realiza sobre un plano de coordenadas cartesianas, ubicando el tiempo en el eje de las abscisas (eje x) y la distancia en el eje de las ordenadas (eje Y)
- Se deberán registrar todas las distancias que se obtengan de los distintos ploteos versus el minuto al cual correspondan. Estos puntos generarán una recta al ser unidos convenientemente, lo cual nos permitirá obtener una estimación de la distancia al blanco por efecto del filtraje realizado. Nota.- Será siempre recta, o al menos recta por secciones, si la velocidad en la línea de mira (ILOS p) permanece casi constante, esto se logrará adoptando rumbos de conducción de un valor aproximadamente constante (por ejemplo de 70° en ambos sentidos de la dirección de movimiento del

blanco, a menos que el blanco efectúe un ZIG y varíe la razón de distancia).

- Cada marca de distancia registrada en el ploteo debe ser convenientemente identificada con la simbología apropiada.
- En la parte inferior del ploteo debe registrarse el aporte de la disminución de distancia originada por el buque propio, debiendo hacerlo cada 15 minutos o cada vez que se cambie de rumbo.
- Cuando se anuncia un posible ZIG del blanco el ploteador Tiempo Distancia deberá registrarlo en su ploteo por medio de una raya la letra Z. Esto le permitirá resaltar que la razón de distancia puede variar a partir de este momento.
- Para cada pierna propia se computa el cálculo de la distancia en la línea de mira por medio de la siguiente fórmula:

$$\text{DISTANCIA} = \frac{K \times \text{ILOS}_p \times t}{60}$$

Donde: t = tiempo en la pierna
K = 2,000 yardas.

CAPÍTULO III

3. Análisis y determinación de requisitos del sistema.

3.1. Definición del Problema

Se requiere la confección de un sistema de control de tiro para Torpedos a bordo de Submarinos de la Marina de Guerra del Perú, ejecutable en Computadoras Personales (P.C.), que sirva de sistema alternativo a la solución de datos del Blanco y datos iniciales de Torpedos.

En la actualidad existe una Computadora de Control de Tiro para dos (02) operadores especialmente diseñada para tal fin, y el sistema alternativo existente para casos de emergencia y/o fallas, se obtiene manualmente mediante procedimientos y ploteos que consumen gran cantidad de tiempo con un alto porcentaje de error.

El objetivo de este sistema es permitir obtener la solución de datos del blanco y datos iniciales de torpedos de manera alterna, integrando los diferentes procedimientos y ploteos del TEAM de ataque.

3.2. Justificación del sistema

Un software que permita obtener la solución de datos del blanco y datos iniciales del torpedo en una computadora personal (P.C.), en

forma rápida, amigable y confiable, elevaría la eficiencia del TEAM de ataque.

La implementación de un sistema de control de tiro alterno computarizado permitiría el ahorro de horas/hombre, llegando a la solución del problema de tiro en menor tiempo y mayor exactitud.

En unidades submarinas antiguas podría convertirse en un sistema de primer orden, debido a que las computadoras a bordo son analógicas de generaciones anteriores (construías mediante sincros, cabezales, engranajes, relays), obteniendo estas últimas, una solución con un gran margen de error debido a su tecnología obsoleta.

Foto N° 3.1: Torpedo Data Computer (TDC) de tecnología obsoleta.



3.3. Objetivos

3.3.1. Objetivos cualitativos para el Producto

- Simplificar los procedimientos y ploteos durante las Fases de Aproximación y Ataque del Submarino.
- Permitir una cómoda y rápida solución del problema de Control de Tiro en Submarinos.

3.3.2. Objetivos cuantitativos para el Producto

- Reducir al mínimo el tiempo para la obtención de una solución en el problema de control de tiro.

3.3.3. Objetivos Cualitativos para el Proceso

- La implementación de una técnica de programación estructurada, simplificará e incrementará la eficiencia del procesamiento.

3.3.4. Objetivos Cuantitativos para el Proceso

- Deberá minimizarse el tiempo de desarrollo.

3.4. Estrategias

Para la elaboración del software se dispone de una gran variedad de herramientas de programación. De todas ellas se escogió el Lenguaje C, debido a que es un lenguaje orientado a resolver problemas gráficos con una amigable interfaz del usuario, y permite construir un ejecutable eficiente en computadoras personales PC.

Se ha considerado un periodo de prueba con los Team de Ataque de los submarinos, a fin de efectuar diversas simulaciones que comprueba la bondad del nuevo sistema, comparando resultados con el sistema obsoleto actual y los resultados reales de los buques blanco.

3.5. Instrumentos y técnicas a usar.

- Metodología de Análisis y Diseño: Yourdon
- Método de Programación: Estructurado
- Técnicas de Prueba: Prueba de la caja negra

3.6. Alcance

El software podrá ser utilizado por las dotaciones de los submarinos de la Marina de Guerra del Perú encargadas de la aproximación y ataque a las Unidades Blanco

3.7. Especificaciones de requisito del sistema

La información y ploteos efectuados por el sistema, esta limitada a la proporcionada en los manuales técnicos de la Fuerza de Submarinos del Perú, a los principales ploteos utilizados en sus unidades y a la experiencia de su personal.

3.7.1. Requerimiento Cuantitativo del Sistema

- La operación del sistema debe de estar disponible a cualquier hora del día.
- El sistema deberá obtener la solución del problema de

tiro hasta un máximo de tres (03) blancos.

3.7.2. Requerimiento Cualitativo del Sistema

- El usuario tendrá acceso al sistema de manera amigable mediante menús tipo POPUP.
- Salida de impresión a elección del usuario.
- El Sistema debe dar respuestas en tiempo real.
- Deberá entregar los principales planteos utilizados en la Fuerza de Submarinos del Perú.

3.7.3. Requerimiento de hardware

- Una computadora personal PC-486 SX o superior con 2 MB de memoria RAM, IMB en el disco duro, impresora de carro angosto, monitor SVGA y teclado.

3.7.4. Requerimiento de software

- Software Base: D.O.S. versión 5.0 o superior.
- Software de Desarrollo: Turbo C.

3.7.5. Requerimientos de Publicaciones

- Manuales técnicos de la Fuerza de Submarinos, baterías de problemas de tiro resueltos existentes en la Escuela de Submarinos y textos de programación.

3.7.6. Criterio de aceptación

Para verificar el cumplimiento de los requerimientos, se efectuarán pruebas en muelle, con simuladores y problemas resueltos, y en la mar durante operaciones con otras unidades

Los niveles de aceptación del sistema serán los siguientes:

- La solución del problema de ataque deberá resolverse dentro de la 15,000 yardas hasta las 4000 yardas, excepto lanzamientos de emergencia.
- El software deberá desarrollar, consolidar y comparar al menos dos (02) de los ploteos manuales utilizados a bordo.
- Durante la fase de aproximación, el sistema deberá recomendar al Oficial Atacante acciones futuras, basándose en la teoría de tradicional de la Táctica Submarina y en experiencias propias de la dotación.
- +/- 1.0 grados de diferencia en la solución de Rumbo del Blanco (RB).
- +/- 100 yardas de diferencia en la solución de Distancia del Blanco (DB).
- +/- 1.0 nudos de diferencia en la solución de Rumbo del Blanco (RB)
- Ploteo simultáneo de tres (03) blancos.
- Ingreso constante de datos durante cinco (05) horas, sin ocasionar fallas.

CAPÍTULO IV

4. Diseño del sistema.

4.1. Subsistemas componentes.

4.1.1. Hardware

Estará compuesto por una computadora personal Pentium o superior, con al menos 156 MB de memoria RAM, 1 MB en disco duro o disquettes de 1.44 MB, impresora, monitor SVGA y teclado.

4.1.2. Software

El software utilizará sistema operativo monousuario Windows 98 o superior.

4.2. Subsistema de interfaz con el usuario

4.2.1. Entorno visual con el usuario

Interfaz amigable con el usuario, apoyado en menús. El usuario libera menús, absolviendo preguntas e ingresando datos constantemente. Asimismo, presentará la selección de ayuda con las principales características de los ploteos ejecutados.

Gráfico 4.1: Entorno visual con el usuario



4.2.2. Entrada y validación de datos

El operador del sistema ingresará los datos en forma manual por intermedio del teclado, la pantalla del monitor mostrará los campos y datos ingresados. El software validará en forma automática si el dato ingresado se encuentra dentro de los parámetros de valor.

4.3. Subsistema de procesamiento

4.3.1. Módulo de Ingreso

Permitirá ingresar los datos iniciales el blanco, los datos de traqueo del blanco durante el problema de tiro como marcación verdadera (MV), distancia al blanco (DB), velocidad del blanco (VB), ángulo en la proa (AP), rumbo propio (RP) y velocidad propia (VP).

Asimismo, permitirá ingresar operaciones de diferentes métodos para obtener distancia en forma pasiva por medio de menús tipo POPUP amigablemente.

4.3.2. Módulo de Proceso de Distancia.

Una vez ingresada la opción del tipo de método, efectuará el cálculo de distancia inicial de acuerdo al teorema del método seleccionado, debiendo indicar instrucciones de cambios de Rumbo y Velocidad.

4.3.3. Módulo de Proceso de Diagrama en la Línea de Mira (LOS)

Graficará los vectores de velocidad del buque propio y del blanco, debiendo efectuar el análisis de los Teoremas de "Diagramas en la Línea de Mira" (LOS por su sigla en Inglés).

4.3.4. Módulo de Proceso de "Ploteo Tiempo Marcación" (TBP)

Graficará un plano de coordenadas cartesianas ubicando las marcaciones en el eje de las abscisas (eje x) y el tiempo en el eje de ordenadas (eje y), debiendo efectuar los análisis de acuerdo al teorema del ploteo Barnard o ploteo TBP.

4.3.5. Módulo de proceso de "Ploteo Geográfico"

Presentará la situación geográfica del problema, representando gráficamente la posición, rumbo y velocidad del blanco y buque propio.

4.3.6. Módulo de Salida de "Instructor de Ataque"

Integrará y analizará las soluciones de los módulos de proceso de los Teoremas, indicando las instrucciones generales de ataque como mejor profundidad de escucha, rumbo de aproximación, rumbo de conducción, distancia a la Derrota, ZIG - ZAG del blanco, preparar y alistar tubos, pasando por la plana y corte, rumbo de alejamiento y mejor profundidad de evasión.

4.3.7. Módulo de Salida General

Presentará en forma de texto en una sola pantalla, las soluciones generales para todos los blancos seleccionados, y el ángulo de giro correspondiente para todos los torpedos.

4.4. Características del usuario

El producto está orientado al personal de oficiales calificados en Submarinos, requiriendo conocimientos de táctica y ploteos submarinos.

4.5. Diccionario de términos

Angulo de Conducción (AC): Es el ángulo formado entre la marcación verdadera al blanco y el rumbo propio.

Angulo de Giro (AG): Es el ángulo medido entre el eje longitudinal del Tubo Lanza Torpedo (TLT) y el rumbo del torpedo.

Angulo en la Proa (AP): Es el ángulo comprendido entre el rumbo del blanco y la recíproca de la marcación verdadera al blanco. Se mide de 0 a 180 grados por babor o estribor.

Rumbo de Aproximación (RA): Es el rumbo del buque propio a ángulo recto con la marcación al blanco y en sentido de acercamiento a este.

Rumbo de Alejamiento (RA): Es el rumbo del buque propio perpendicular a la derrota del blanco en el sentido de alejarse a éste.

Rumbo Propio (RP): Es el rumbo al cual navega el buque propio.

Rumbo del Blanco (RB): Es el rumbo al cual navega el buque blanco.

Marcación Verdadera (MV): Es el ángulo comprendido entre el norte verdadero y la marcación del blanco.

Turbo C: Lenguaje de programación estructurada fabricada por la Borland INT.

Versión Prototipo: Primera versión del programa en borrador y poco amigable, utilizada con la finalidad de efectuar pruebas.

4.6. Especificación del Plan de Pruebas

Con la asesoría de un experto en Táctica Avanzada para Submarinos se preparará una batería de prueba dentro de las restricciones y especificaciones del sistema, debiendo ejecutarse con los simuladores de ataque existentes a bordo de los submarinos tipo 209, y complementarse al menos con un (01) ejercicio en la mar.

Para efectos de la entrega de la versión prototipo del sistema, se han efectuado diversas pruebas del banco de ejercicios existentes en la Escuela de Submarinos, con resultados satisfactorios. A continuación se grafican tres ejemplos de solución del sistema con datos en situaciones diferentes:

Gráfico 4.2: Entrada y validación de datos Ejemplo N° 1.



Gráfico 4.3: Solución Ejemplo N° 1.



Gráfico 4.4: Entrada y validación de datos Ejemplo N° 2.



Gráfico 4.5: Solución Ejemplo N° 2.



Gráfico 4.6: Entrada y validación de datos Ejemplo N° 3.



Gráfico 4.2: Solución Ejemplo N° 3.



CAPÍTULO V

5. Codificación.

```
#include <stdio.h>
#include <conio.h>
#include <graphics.h>
#include <math.h>
typedef int BOOLEAN;

float rp,mv,mvt,ac,vp,alosp,alospb,alosp,ilosb,
      rb,rbm,rbs,rblm,rbt,rac,raprox,rconduc,ral,
      ap,apm,aps,apl,aptbp,apcin,apgeo,apt, /****en el prog solo con
ap**/
      vb,vbm,vbs,vblm,vbtp,vbcin,vbgeo,vbt,
      da,dam,das,dalm,datbp,dacin,dageo,dat,datel,
      b,convrad(float g),convgrad(float r),unidadlm,dd,
      distdd,distpreparar,distalistar,
      napa,profevasion,profescucha;

int bandac,bandap,dirb,xmax,x1,x2,x3,x4,ymax,y1,y2,y3,xb,yb,ybr;

BOOLEAN zz;

main()
{
    clrscr();
```

```

inigraf();
pantalla();
inidatini();
inidatman();   /***aqui es el bucle***/
ingresadatos();
erroringreso();
comparadatos();
lineamira();
graflineamira();
grafcinematica();
compararesultadosap();
compararesultadosvb();
compararesultadosda();
grafgeografico();
graftbp();
grafinstructor();
getch();
/****clrscr();***/
}

```

```

inigraf()
{
    int g_driver,g_mod0,g_error;
    detectgraph(&g_driver,&g_mod0);
    initgraph(&g_driver,&g_mod0, "c:turboc\include");
    g_error = graphresult();

    if (g_error<0)

```

```

    {printf("Error en la inicializaci n de gr ficos:%s\n",
        grapherrormsg(g_error));
    exit(1);
    }
}

```

pantalla()

```

{
    xmax=getmaxx()-8;
    ymax=getmaxy()-16;
    x1=4;                /****** (alt 227)******/
    x2=4+xmax/3;
    x3=4+2*xmax/3;
    x4=4+xmax;
    y1=4;
    y2=4+ymax/2;
    y3=4+ymax;
    setcolor(10);
    settextstyle(0,0,1);
    outtextxy(x1+4,y3+2,"F1      F2      F3      F6      F8");
    setcolor(2);
    outtextxy(x1+4,y3+2," -Contacto A  -Contacto B  -Contacto C  -
Escala  -Imprimir");
    rectangle(x1,y1,x2,y2);
    rectangle(x1,y2,x2,y3);
    rectangle(x2,y1,x3,y2);
    rectangle(x2,y2,x3,y3);
    rectangle(x3,y1,x4,y2);
}

```

```

rectangle(x3,y2,x4,y3);
setcolor(14);
settextstyle(0,0,1);
outtextxy(x3+4,y1+3," INGRESO DE DATOS");
outtextxy(x3+4,y1+4," _____");
outtextxy(x2+4,y1+3," CINEMATICA");
outtextxy(x2+4,y1+4," _____");
outtextxy(x1+4,y1+3," LINEA DE MIRA");
outtextxy(x1+4,y1+4," _____");
outtextxy(x1+4,y1+15," BLANCO");
outtextxy(x1+4,y2-14," BUQUE PROPIO");
outtextxy(x3+4,y2+3," INSTRUCTOR DE ATAQUE");
outtextxy(x3+4,y2+4," _____");
outtextxy(x1+4,y2+3," GEOGRAFICO");
outtextxy(x1+4,y2+4," _____");
outtextxy(x3+4,y1+14," PROPIOS");
outtextxy(x3+4,y1+37," BLANCO");
outtextxy(x3+4,y1+59," MANUAL ");
setcolor(22);
outtextxy(x3+4,y1+59," SISTEMA");
setcolor(2);
outtextxy(x3+4,y1+25," RP= __ VP= __ NAPA= __");
outtextxy(x3+4,y1+48," MV= __ MINUTO= B§=");
outtextxy(x3+4,y1+70," DIST= _____ yd");
outtextxy(x3+4,y1+81," AP= __ BR(_) ER(_)");
outtextxy(x3+4,y1+92," VB= __");
outtextxy(x3+4,y1+103," RB= __");
outtextxy(x3+4,y1+114," ZIG-ZAG(_)");
outtextxy(x3+4,y1+125," ACERC(_) ALEJ(_) CONST(_)");

```

```

outtextxy(x3+4,y1+136," DISTANCIA INICIAL(_));
outtextxy(x3+4,y1+147," DISTANCIA TELEMETRICA(_));
outtextxy(x3+4,y1+158," SELECCION ARMA(_));
setcolor(3);
setttextstyle(0,0,2);
outtextxy(x2+15,y2-17," CONTACTO A");
setttextstyle(0,0,1);
setcolor(2);
}

```

inidatini()

```

{
  rp=-1; vp=-1; mv=-1; b=-1; bandap=-1; bandac=-1; dirb=-1;*****falta
  ingreso b()*****/
}

```

inidatman()

```

{
  mv=-1; ac=-1; zz=0;
  alosp=-1; alosb=-1; alosr=-1; ilosp=-1; ilosb=-1;
  rb=-1; rbm=-1; rbs=-1; rbt=-1;
  ap=-1; apm=-1; aps=-1; aplm=-1; aptbp=-1; apcin=-1; apgeo=-1; apt=-1;
  vb=-1; vbm=-1; vbs=-1; vblm=-1; vbtbp=-1; vbcin=-1; vbgeo=-1; vbt=-
  1;
  da=-1; dam=-1; das=-1; dalm=-1; datbp=-1; dacin=-1; dageo=-1; dat=-1;
  napa=-1;
}

```

ingresadatos()

```
{  
rp=47; vp=6; mv=004; vbt=14; dat=8000; b=1.2; dirb=2; napa=100;  
/****1.9*****/  
/***rp=90; vp=0; mv=000; apt=45; bandap=2; dat=10000; b=1.5;  
/****1.6*****/  
/***rp=90; vp=6; mv=182; rbt=64; vbt=16; b=2.1; dirb=1;  
/****1.10*****/  
/***if zz, zz=1****/  
seleccionarma();  
}
```

seleccionarma()

```
{/****para mk14****/  
distdd=9000; distpreparar=8000; distalistar=4500;}
```

erroringreso()

```
{  
if ((rp<000) || (rp>359.99)) mensagerrorrp();  
if ((vp<0) || (vp>25)) mensagerrorvp();  
if ((mv<-1) || (mv>359.99)) mensagerrormv();  
if ((apt<-1) || (apt>180)) mensagerrorap(); else (apm=apt);  
if ((dat<-1) || (dat>50000)) mensagerrorda(); else (dam=dat);  
if ((vbt<-1) || (vbt>90)) mensagerrorvb(); else (vbm=vbt);  
if ((rbt<-1) || (rbt>359.99)) mensagerrorrb(); else (rbm=rbt);
```

```
if (rbm!=-1) conv_rb_ap();  
}
```

```
mensagerrrrp()  
{/**error en el ingreso de datos del rumbo propio**/}
```

```
mensagerrrvp()  
{/**error en el ingreso de velocidad propia**/}
```

```
mensagerrrmv()  
{/**error en el ingreso de marcación verdadera**/}
```

```
mensagerrrap()  
{/**error en el ingreso de angulo en la proa**/}
```

```
mensagerrorda()  
{/**error en el ingreso de distancia actual**/}
```

```
mensagerrrvb()  
{/**error en el ingreso de velocidad del blanco**/}
```

```
mensagerrrrb()
```

```
{/****error en ingreso de rumbo del blanco****/}
```

```
****
```

```
leap()
```

```
{
```

```
if ((apm==180) || (apm==00)) {bandap=0}
```

```
if cursor=ER {bandap=2}
```

```
if cursor=BR {bandap=1}
```

```
}
```

```
****/
```

```
sucedezz()
```

```
{/****aqui todos los datos -da=-1 al final zz=0 ponerlo debajo de compara  
datos****/}
```

```
comparadatos()
```

```
{
```

```
if ((apm!=-1) && (aps==-1)) ap=apm;
```

```
if ((apm!=-1) && (aps!=-1)) ap=(0.5*apm)+(0.5*aps); /****if zz aps=-  
1,ap=-1****/
```

```
if ((apm==-1) && (aps!=-1)) ap=aps;
```



```

if ((dam!=-1) && (das==-1)) da=dam;
if ((dam!=-1) && (das!=-1)) da=dam;/**tenemos que castigar dam a
su ingreso**/
if ((dam==-1) && (das!=-1)) da=das;
if ((vbm!=-1) && (vbs==-1)) vb=vbm;
if ((vbm!=-1) && (vbs!=-1)) vb=(0.5*vbm)+(0.5*vbs);/**if zz vbs=-
1,vb=-1***/
if ((vbm==-1) && (vbs!=-1)) vb=vbs;
if (zz==1) sucedezz();
}

```

bandapdirb()

```

{
if ((bandap==-1) && (dirb!=-1)) bandap=dirb;
}

```

lineamira()

```

{
bandapdirb();
conv_rp_ac();
if ((ap!=-1) && (da!=-1) && (vb!=-1))
{
alosp=vp*sin(convrad(ac));
ilosp=vp*cos(convrad(ac));
alospb=vb*sin(convrad(ap));
ilosb=vb*cos(convrad(ap));
if (bandac==bandap) alosp=alospb-ilosp;
}
}

```

```

    if (bandac!=bandap) alosr=alosp+alosp;
}
if ((ap!=-1) && (vb!=-1) && (da==1))
{
    alosp=vp*sin(convrad(ac));
    ilosp=vp*cos(convrad(ac));
    alosb=vb*sin(convrad(ap));
    ilosb=vb*cos(convrad(ap));
    if (bandac==bandap) alosr=alosp-alosp;
    if (bandac!=bandap) alosr=alosp+alosp;
    if (b>0) { dalm=(alosp*1936)/b; da=dalm;}
    if (alosp==0) { dalm=(alosp*1936)/b; da=dalm;}
}
if ((vb!=-1) && (da!=1) && (ap==1))
{
    alosp=vp*sin(convrad(ac));
    alosr=(da*b)/1936;
    alosb=alosp+alosp;
    ap=convgrad(asin(alosp/vb));
    aplm=ap;
    ilosp=vp*cos(convrad(ac));
    ilosb=vb*cos(convrad(ap));
}
if ((ap!=-1) && (da!=1) && (vb==1))
{
    alosp=vp*sin(convrad(ac));
    alosr=(da*b)/1936;
    alosb=alosp+alosp;
    vb=alosp/sin(convrad(ap));
}

```

```

vblm=vb;
ilosp=vp*cos(convrad(ac));
ilosb=vb*cos(convrad(ap));
}
if ((ap!=-1) && (vb==-1) && (da==-1))
{
  if (b==0)
  {
    alosp=vp*sin(convrad(ac));
    ilosp=vp*cos(convrad(ac));
    vb=alosp/sin(convrad(ap));
    vblm=vb;
    alosb=vb*sin(convrad(ap));
    ilosb=vb*cos(convrad(ap));
    if (bandac==bandap) alosr=alosp-aalosb;
    if (bandac!=bandap) alosr=alosp+alosp;
    if (b>0) { da=(alosr*1936)/b; dalm=da;}
    if (alosp==0) { da=(alosp*1936)/b; dalm=da;}
  }
}
if ((vb!=-1) && (ap==-1) && (da==-1))
{
  if (b==0)
  {
    alosp=vp*sin(convrad(ac));
    ilosp=vp*cos(convrad(ac));
    ap=convgrad(asin(alosp/vb));
    apm=ap;
    alosb=vb*sin(convrad(ap));

```

```

ilosb=vb*cos(convrad(ap));
if (bandac==bandap) alosr=alosp;
if (bandac!=bandap) alosr=alosp;
if (b>0) {da=(alosp*1936)/b; dalm= da;}
if (alosp==0) {da=(alosp*1936)/b; dalm=da;}
}
}
if ((vb==1) && (ap==1) && (da!=1))
{
    alosp=vp*sin(convrad(ac));
    alosr=(da*b)/1936;
    alosb=alosp; *****/
}
/**** grafileamira(); *****/
}

```

conv_rp_ac()

```

{
while(1)
{
if ((rp-mv==180) || (mv-rp==180)) {ac=180; bandac=0; break;}
if ((rp-mv==000) || (mv-rp==000)) {ac=000; bandac=0; break;}
if ((rp<180) && (mv<180) && (rp>mv)) {ac=rp-mv; bandac=2; break;}
if ((rp<180) && (mv<180) && (mv>rp)) {ac=mv-rp; bandac=1; break;}
if ((rp>180) && (mv<180) && (rp>mv)) {ac=(360-rp)+mv; bandac=1;
break;}
if ((rp>000) && (mv==000) && (rp<180)) {ac=rp; bandac=2; break;}
if ((rp<180) && (mv>180) && (mv>rp)) {ac=(360-mv)+rp; bandac=2;

```

```

break;}
    if ((rp>180) && (mv>180) && (rp>mv)) {ac=rp-mv; bandac=2; break;}
    if ((rp>180) && (mv==000) && (rp<360)) {ac=360-rp; bandac=1;
break;}
    if ((rp>180) && (mv>180) && (mv>rp)) {ac=mv-rp; bandac=1; break;}
    }
    }

```

```

conv_rb_ap()
{
    float temp;
    if ((mv>=180) && (mv<360)) temp=mv-180;
    if ((mv<180) && (mv>=0)) temp=mv+180;
    if (temp>rbm) {ap=temp-rbm; bandap=2;}
    if (temp<rbm) {ap=rbm-temp; bandap=1;}
}

```

```

conv_ap_rb()
{
    float temp;
    if ((mv>=180) && (mv<360)) temp=mv-180;
    if ((mv<180) && (mv>=0)) temp=mv+180;
    if (bandap==1) rbs=aps+temp;
    if (bandap==2) rbs=temp-aps;
}

```

```

graflineamira()
{ /**float unidadlm;****/
    int lineamax,xac,yac,xap,yap,x0;
/****ilosb=14.1; ilosp=12.2; vp=4; ac=30; bandac=2; b=20; alosr=5; vb=15;
ap=30; bandap=1; *****/
    blanco(x1,y1);
    line((x1+x2)/2,y1+34,(x1+x2)/2,y2-34); getch();
    if ((alosb!=-1) && (alosp!=-1))
    {
        x0=(x1+x2)/2;
        setcolor(10);
        unidadlm=((y2-20)-
(y1+24))/(ilosp+ilosb+(alosr/tan(convrad(b)))/(da/1000)));
        if (bandac==2)
        {
            line((x1+x2)/2,y2-
34,(((x1+x2)/2)+(unidadlm*vp*sin(convrad(ac))),(y2-34)-
(unidadlm*vp*cos(convrad(ac)))); getch();
            xac=(((x1+x2)/2)+(unidadlm*vp*sin(convrad(ac))));
            yac=((y2-34)-(unidadlm*vp*cos(convrad(ac))));
        }
        if ((bandac==1) || (bandac==0))
        {
            line((x1+x2)/2,y2-34,(((x1+x2)/2)-
(unidadlm*vp*sin(convrad(ac))),(y2-34)-
(unidadlm*vp*cos(convrad(ac)))); getch();
            xac=(((x1+x2)/2)-(unidadlm*vp*sin(convrad(ac))));
            yac=((y2-34)-(unidadlm*vp*cos(convrad(ac))));
        }
        if ((bandap==2) && (ap!=-1))

```

```

{
line((x1+x2)/2,y1+34,(((x1+x2)/2)+(unidadlm*vb*sin(convrad(ap))),(y1
+24)+(unidadlm*vb*cos(convrad(ap)))); getch();
    xap=(((x1+x2)/2)+(unidadlm*vb*sin(convrad(ap))));
    yap=((y1+24)+(unidadlm*vb*cos(convrad(ap))));
}
if ((bandap==1) && (ap!=-1))
{
    line((x1+x2)/2,y1+34,(((x1+x2)/2)-
(unidadlm*vb*sin(convrad(ap))),(y1+24)+(unidadlm*vb*cos(convrad(a
p)))); getch();

        xap=(((x1+x2)/2)-(unidadlm*vb*sin(convrad(ap))));
        yap=((y1+24)+(unidadlm*vb*cos(convrad(ap))));
    }
    if ((bandap==0) && (ap!=-1))
    {
        line((x1+x2)/2,y1+34,(((x1+x2)/2)-
(unidadlm*vb*sin(convrad(ap))),(y1+24)+(unidadlm*vb*cos(convrad(a
p)))); getch();

            xap=(((x1+x2)/2)-(unidadlm*vb*sin(convrad(ap))));
            yap=((y1+24)+(unidadlm*vb*cos(convrad(ap))));
        }

        line(xap,yap,xac,yac); getch();
    if ((bandap==1) || (bandap==0))
    {
        outtextxy(x1+20,yap,"Vb");
        outtextxy(x1+20,((yac+yap)/2),"DA");
    }
}

```

```

}
if (bandap==2)
    {
        outtextxy(x2-60,yap,"Vb");
        outtextxy(x2-60,(yac+yap)/2,"DA");
    }
if ((bandac==1) || (bandac==0)) outtextxy(x1+20,yac,"Vp");
if (bandac==2) outtextxy(x2-60,yac,"Vp");
setcolor(2);
line(xac,yac,xac,yap); getch();
line(xap,yap,xac,yap); getch();
line(xac,yac,x0,yac); getch();
line(xap,yap,x0,yap); getch();
}
}

```

grafgeografico()

```

{
    blanco(x1,y2);
}

```

grafbp()

```

{
    blanco(x2,y2);
    settextstyle(0,1,1);
    setcolor(14);
    outtextxy(x2+15,y2+70," T I E M P O");
}

```



```

settextstyle(0,0,1);
outtextxy(x2+4,y3-30,"    M V");
outtextxy(x2+4,y2+3,"    T B P");
outtextxy(x2+4,y2+4,"    _____");
setcolor(2);
line(x2+19,y2+34,x2+19,y3-34);
line(x2+19,y3-34,x3-19,y3-34);
}

```

```

blancores(ybr)
{ setcolor(0);
  outtextxy(x4-42,y1+ybr,"    ÔÔÔÔ
");
  setcolor(22);
}

```

```

blanco(xb,yb)
{ setcolor(0);
  outtextxy(xb+4,yb+25,"    ÔÔÔÔ
");
  outtextxy(xb+4,yb+30,"    ÔÔÔÔ
");
  outtextxy(xb+4,yb+35,"    ÔÔÔÔ
");
  outtextxy(xb+4,yb+40,"    ÔÔÔÔ
");
  outtextxy(xb+4,yb+45,"    ÔÔÔÔ
");
}

```

```
");
    outtextxy(xb+4,yb+50,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+55,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+60,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+65,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+70,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+75,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+80,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+85,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+90,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+95,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+100,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+105,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+110,"ÛÛÛÛ
");
    outtextxy(xb+4,yb+115,"ÛÛÛÛ
");
```

outtextxy(xb+4,yb+120,"ÛÛÛÛ");
outtextxy(xb+4,yb+125,"ÛÛÛÛ");
outtextxy(xb+4,yb+130,"ÛÛÛÛ");
outtextxy(xb+4,yb+135,"ÛÛÛÛ");
outtextxy(xb+4,yb+140,"ÛÛÛÛ");
outtextxy(xb+4,yb+145,"ÛÛÛÛ");
outtextxy(xb+4,yb+150,"ÛÛÛÛ");
outtextxy(xb+4,yb+155,"ÛÛÛÛ");
outtextxy(xb+4,yb+160,"ÛÛÛÛ");
outtextxy(xb+4,yb+165,"ÛÛÛÛ");
outtextxy(xb+4,yb+170,"ÛÛÛÛ");
outtextxy(xb+4,yb+175,"ÛÛÛÛ");
outtextxy(xb+4,yb+180,"ÛÛÛÛ");
outtextxy(xb+4,yb+185,"ÛÛÛÛ");
outtextxy(xb+4,yb+190,"ÛÛÛÛ");

```
");
    outtextxy(xb+4,yb+195,"ÛÛÛÛ");
");
    outtextxy(xb+4,yb+200,"ÛÛÛÛ");
");
    outtextxy(xb+4,yb+204,"ÛÛÛÛ");
");
    setcolor(2);
}
```

```
float convrad(float g)
{
    return((g*3.141592)/180);
}

float convgrad(float r)
{
    return(180*r/3.141592);
}
```

```
grafcinematica()
{
    blanco(x2,y1);
    circle((x2+x3)/2,(y1+y2)/2,((x3-x2)/2)-20);
}
```

```
grafinstructor()
```

```

{
  setcolor(14); /***amarillo***/
  outtextxy(x3+4,y2+15,"MEJOR PROF.ESCUCHA p.");
  outtextxy(x3+4,y2+26,"RUMBO APROXIMACION =");
  rumboaproximacion();
  outtextxy(x3+4,y2+37,"RUMBO CONDUCCION =");
  rumboconduccion();
  outtextxy(x3+4,y2+48,"MARCACION AL PMA =");
  marcacionpma();

  outtextxy(x3+4,y2+59,"DIST. DERROTA (DD) =");
  distanciaderrota();
  /*** outtextxy(x3+4,y2+103,"POSIBLE ZIG-ZAG BLANCO");*****/
  setcolor(14);
  preparartubos();
  outtextxy(x3+4,y2+136,"PASANDO POR EL CORTE");
  setcolor(4);/*******de alerta*****/
  outtextxy(x3+4,y2+147,"PASANDO POR LA PLANA");
  outtextxy(x3+4,y2+158,"ARMA SELECCIONADA");
  outtextxy(x3+4,y2+169,"EL SISTEMA TIENE SOLUCION");
  outtextxy(x3+4,y2+180,"ANGULO DE GIRO MK14");
  outtextxy(x3+4,y2+191,"RUMBO DE ATAQUE");
  outtextxy(x2+4,y2+213," ACERCANDOSE"); /***ALEJANDOSE
- RUMBO COLISION***/
  setcolor(14);
  outtextxy(x3+4,y2+202,"RUMBO ALEJAMIENTO =");
  rumboalejamiento();
  outtextxy(x3+4,y2+213,"MEJOR PROF.EVASION =");
  mejorprofevasion();

```

```

    outtextxy(x3+4,y2+225,"F7 PARA RECOMENDAC. MK 37");
    setcolor(2);
}

```

```

mejorprofevasion()
{char texto[17];
  if (napa!=-1)
    {if (napa<201) profevasion=30*sqrt(napa);
      if (napa>200) profevasion=napa+200;
      setcolor(22);      blancos(y2+213);sprintf(texto,"%f",profevasion,"
p.");outtextxy(x4-42,y2+213,texto);
      setcolor(14);
    }
}

```

```

rumboaproximacion()
{char texto[17];
  if (bandap==1)
    { raprox=mv-90; if (raprox>360){raprox=raprox-360;}; if (raprox<0)
{raprox=raprox+360;};}
  if (bandap==2)
    { raprox=mv+90; if (raprox>360){raprox=raprox-360;}; if (raprox<0)
{raprox=raprox+360;};}
  if      (raprox!=-1){setcolor(22);blancos(y2+26);sprintf(texto,"%-
03.1f",raprox);outtextxy(x4-42,y2+26,texto);
    setcolor(14);}
}

```

```

rumboconduccion()
{char texto[17];
  float angconduc;
  angconduc=2*aps;
  if (angconduc>70.1) {angconduc=70;}
  if (bandap==1)
    {rconduc=mv-angconduc; if (rconduc>360){rconduc=rconduc-360;}; if
(rconduc<0) {rconduc=rconduc+360;};}
  if (bandap==2)
    {rconduc=mv+angconduc; if (rconduc>360){rconduc=rconduc-360;};
if (rconduc<0) {rconduc=rconduc+360;};}
  if (rconduc!=-1){setcolor(22);blancores(y2+37);sprintf(texto,"%-
03.1f",rconduc);outtextxy(x4-42,y2+37,texto);
  setcolor(14);}
}

```

```

marcacionpma()
{
  char texto[17];
  float anguloacerc;
  if (aps!=-1){anguloacerc=90-aps;
  if (bandap==1)
    { rac=mv-anguloacerc; if (rac>360){rac=rac-360;}; if (rac<0)
    {rac=rac+360;};}
  if (bandap==2)

```

```

    { rac=mv+anguloacerc; if (rac>360){rac=rac-360;}; if (rac<0)
    {rac=rac+360;};}
    setcolor(22);blancores(y2+48);sprintf(texto,"%-
03.1f",rac);outtextxy(x4-42,y2+48,texto);}
    /****setcolor(0);outtextxy(x3+4,y2+48,"ÛÛÛÛ
");
    if (vbs!=-1)
        if (bandap==1)
            { rac=mv-anguloacerc;
            if (bandap==2)
                {rconduc=mv+angconduc;
                {dpma = 1936*(vbs+-alosp)/b;
                setcolor(22);
                sprintf(texto,"%f",dpma);outtextxy(x3+60,y2+48,texto);
                setcolor(14); outtextxy(x3+4,y2+48,"Mv.PMA =   D.PMA =");
                }*****/
    setcolor(14);
    }

```

```

distanciaderrota()
{
    char texto[20];
    if ((aps!=-1)&& (das!=-1)) {dd=das*sin(convrad(aps));
        blancores(y2+59);sprintf(texto,"%f",dd);outtextxy(x4-
42,y2+59,texto);
        if (dd<1000) {setcolor(4);/****de alerta****/
        outtextxy(x3+4,y2+70,"(DD) MUY PEQUEÑA");sonido1();
        outtextxy(x3+4,y2+81,"ABRIRSE DE LA DERROTA");

```



```

    outtextxy(x3+4,y2+92,"(EXCEPTO TIRO GARGANTA)");}
    if (dd>distdd) {setcolor(4);/*****de alerta*****/
    outtextxy(x3+4,y2+70,"(DD) EXCESIVA");sonido1();
    outtextxy(x3+4,y2+81,"PARA ESTRECHAR (DD) ADOPTE");
        outtextxy(x3+4,y2+92,"RUMBO
            ACERCAMIENTO ="); rumboacercamiento();}
    if ((dd>1000) && (dd<distdd)) {setcolor(0);/*****de alerta*****/
    outtextxy(x3+4,y2+70,"ÛÛÛÛ
");
    outtextxy(x3+4,y2+81,"ÛÛÛÛ
");
    outtextxy(x3+4,y2+92,"ÛÛÛÛ
");}
    setcolor(2);}
}

```

```

rumboacercamiento()
{
char texto[17];
float anguloacerc;
if (aps!=-1){anguloacerc=90-aps;
    if (bandap==1)
{   rac=mv-anguloacerc;   if   (rac>360){rac=rac-360;};   if   (rac<0)
    {rac=rac+360;};}
    if (bandap==2)
{   rac=mv+anguloacerc;   if   (rac>360){rac=rac-360;};   if   (rac<0)
    {rac=rac+360;};}
setcolor(22);blancos(y2+92);sprintf(texto,"%-03.1f",rac);outtextxy(x4-

```

```
42,y2+92,texto);}
}
```

```
posiblezz()
{}
```

```
preparartubos()
{
    char texto[20];
    if (das!=-1)
        {if (das<distpreparar) {setcolor(14);/*****amarillo*****/
        outtextxy(x3+4,y2+114,"PREPARAR TUBOS ");}
        if (das<distalistar) {setcolor(14);
        outtextxy(x3+4,y2+125,"ALISTAR TUBOS ");sonido1();}
        if (das>distpreparar) {setcolor(0);/*****negro*****/
        outtextxy(x3+4,y2+114,"ÛÛÛÛ
");}
        if (das>distalistar) {setcolor(0);
        outtextxy(x3+4,y2+125,"ÛÛÛÛ
");}
        setcolor(2);}
}
```

```
pasandocorte()
{}
```

```
pasandoplana()
```

```
{}
```

```
armaseleccionada()
```

```
{}
```

```
sistemasolucion()
```

```
{}
```

```
angulogiro()
```

```
{}
```

```
rumboataque()
```

```
{}
```

```
rumboalejamiento()
```

```
{ char texto[17];
```

```
float anguloacerc;
```

```
if (aps!=-1){anguloacerc=90-aps;
```

```
    if (bandap==1)
```

```
{    ral=mv-anguloacerc+180;  if  (ral>360){ral=ral-360;};  if  (ral<0)
```

```
    {ral=ral+360;};}
```

```
    if (bandap==2)
```

```
{  ral=mv+anguloacerc+180;  if  (ral>360){ral=ral-360;};  if  (ral<0)
  {ral=ral+360;};}
setcolor(22);blancores(y2+202);sprintf(texto,"%-03.1f",ral);outtextxy(x4-
  42,y2+202,texto);}
setcolor(14);
}
```

```
evasion()
{}
```

```
recomendacionmk37()
{}
```

```
sonido1()
{
  sound(200);
  delay(1);
  nosound();
}
```

```
sonido3()
{
  int i;
  i=1;
  while(i<4)
```

```

    {
        sound(440);
        delay(i*150);
        nosound();
        delay(i*100);
        i++;
    }
}

```

compararesultadosap()

```

{
    BOOLEAN apsok;
    char texto[20];
    apsok=0;
    setcolor(22);
    if ((apl!= -1) && (aptbp== -1) && (apcin== -1) && (apgeo== -1))
    {aps=apl;   blancos(81);   sprintf(texto,"%f",aps);   outtextxy(x4-
        42,y1+81,texto);apsok=1;}
    if ((apl== -1) && (aptbp!= -1) && (apcin== -1) && (apgeo== -1))
    {aps=aptbp;   blancos(81);   sprintf(texto,"%f",aps);   outtextxy(x4-
        42,y1+81,texto);apsok=1;}
    if ((apl== -1) && (aptbp== -1) && (apcin!= -1) && (apgeo== -1))
    {aps=apcin;   blancos(81);   sprintf(texto,"%f",aps);   outtextxy(x4-
        42,y1+81,texto);apsok=1;}
    if ((apl== -1) && (aptbp== -1) && (apcin== -1) && (apgeo!= -1))
    {aps=apgeo;   blancos(81);   sprintf(texto,"%f",aps);   outtextxy(x4-
        42,y1+81,texto);apsok=1;}
    if ((apl!= -1) && (aptbp!= -1) && (apcin== -1) && (apgeo== -1))

```

```

{aps=aplm*0.5+aptbp*0.5;    blancos(81);    sprintf(texto,"%f",aps);
    outtextxy(x4-42,y1+81,texto);apsok=1;}
if ((aplm!=-1) && (aptbp==-1) && (apcin!=-1) && (apeo==-1))
{aps=aplm*0.5+apcin*0.5;    blancos(81);    sprintf(texto,"%f",aps);
    outtextxy(x4-42,y1+81,texto);apsok=1;}
if ((aplm!=-1) && (aptbp==-1) && (apcin==-1) && (apeo!=-1))
{aps=aplm*0.5+apeo*0.5;        blancos(81);sprintf(texto,"%f",aps);
    outtextxy(x4-42,y1+81,texto);apsok=1;}
if ((aplm==-1) && (aptbp!=-1) && (apcin!=-1) && (apeo==-1))
    {aps=aptbp*0.5+apcin*0.5;    blancos(81);    sprintf(texto,"%f",aps);
outtextxy(x4-42,y1+81,texto);apsok=1;}
if ((aplm==-1) && (aptbp!=-1) && (apcin==-1) && (apeo!=-1))
{aps=aptbp*0.5+apeo*0.5;        blancos(81);sprintf(texto,"%f",aps);
    outtextxy(x4-42,y1+81,texto);apsok=1;}
if ((aplm==-1) && (aptbp==-1) && (apcin!=-1) && (apeo!=-1))
{aps=apcin*0.5+apeo*0.5;        blancos(81);sprintf(texto,"%f",aps);
    outtextxy(x4-42,y1+81,texto);apsok=1;}
if ((aplm!=-1) && (aptbp!=-1) && (apcin!=-1) && (apeo==-1))
{aps=aplm*0.4+aptbp*0.3+apcin*0.3;
    blancos(81);sprintf(texto,"%f",aps);                outtextxy(x4-
    42,y1+81,texto);apsok=1;}
if ((aplm==-1) && (aptbp!=-1) && (apcin!=-1) && (apeo!=-1))
{aps=aptbp*0.3+apcin*0.4+apeo*0.3;
    blancos(81);sprintf(texto,"%f",aps);                outtextxy(x4-
    42,y1+81,texto);apsok=1;}
if ((aplm!=-1) && (aptbp==-1) && (apcin!=-1) && (apeo!=-1))
{aps=aplm*0.4+apcin*0.3+apeo*0.3;
    blancos(81);sprintf(texto,"%f",aps);                outtextxy(x4-
    42,y1+81,texto);apsok=1;}

```

```

if ((apsok==0) && (aps!=-1)) {aps=ap; blancos(81);
    sprintf(texto,"%f",aps); outtextxy(x4-42,y1+81,texto);}
if (aps!=-1)
{ conv_ap_rb(); if (rbs!=-1){blancos(103); sprintf(texto,"%-03.1f",rbs);
    outtextxy(x4-42,y1+103,texto);}}
    setcolor(2);
}

```

compararesultadosvb()

```

{
    BOOLEAN vbsok;
    char texto[20];
    vbsok=0;
    setcolor(22);
    if ((vblm!=-1) && (vbtbp===-1) && (vbcin===-1) && (vbgeo===-1))
{ vbs=vblm; blancos(92); sprintf(texto,"%f",vbs); outtextxy(x4-
    42,y1+92,texto); vbsok=1;}
    if ((vblm===-1) && (vbtbp!=-1) && (vbcin===-1) && (vbgeo===-1))
{ vbs=vbtbp; blancos(92); sprintf(texto,"%f",vbs); outtextxy(x4-
    42,y1+92,texto);vbsok=1;}
    if ((vblm===-1) && (vbtbp===-1) && (vbcin!=-1) && (vbgeo===-1))
{ vbs=vbcin; blancos(92); sprintf(texto,"%f",vbs); outtextxy(x4-
    42,y1+92,texto);vbsok=1;}
    if ((vblm===-1) && (vbtbp===-1) && (vbcin===-1) && (vbgeo!=-1))
{ vbs=vbgeo; blancos(92); sprintf(texto,"%f",vbs); outtextxy(x4-
    42,y1+92,texto);vbsok=1;}
    if ((vblm!=-1) && (vbtbp!=-1) && (vbcin===-1) && (vbgeo===-1))
{ vbs=vblm*0.5+vbtbp*0.5; blancos(92); sprintf(texto,"%f",vbs);

```

```

    outtextxy(x4-42,y1+92,texto);vbsok=1;}
if ((vblm!=-1) && (vbtbp==1) && (vbcin!=-1) && (vbgeo==1))
{vbs=vblm*0.5+vbcin*0.5;   blancos(92);   printf(texto,"%f",vbs);
  outtextxy(x4-42,y1+92,texto);vbsok=1;}
if ((vblm!=-1) && (vbtbp==1) && (vbcin==1) && (vbgeo!=1))
{vbs=vblm*0.5+vbgeo*0.5;   blancos(92);   printf(texto,"%f",vbs);
  outtextxy(x4-42,y1+92,texto);vbsok=1;}
if ((vblm==1) && (vbtbp!=1) && (vbcin!=-1) && (vbgeo==1))
{vbs=vbtbp*0.5+vbcin*0.5;   blancos(92);   printf(texto,"%f",vbs);
  outtextxy(x4-42,y1+92,texto);vbsok=1;}
if ((vblm==1) && (vbtbp!=1) && (vbcin==1) && (vbgeo!=1))
{vbs=vbtbp*0.5+vbgeo*0.5;   blancos(92);   printf(texto,"%f",vbs);
  outtextxy(x4-42,y1+92,texto);vbsok=1;}
if ((vblm==1) && (vbtbp==1) && (vbcin!=1) && (vbgeo!=1))
{vbs=vbcin*0.5+vbgeo*0.5;   blancos(92);   printf(texto,"%f",vbs);
  outtextxy(x4-42,y1+92,texto);vbsok=1;}
if ((vblm!=-1) && (vbtbp!=1) && (vbcin!=-1) && (vbgeo==1))
{vbs=vblm*0.4+vbtbp*0.3+vbcin*0.3;           blancos(92);
  printf(texto,"%f",vbs); outtextxy(x4-42,y1+92,texto);vbsok=1;}
if ((vblm==1) && (vbtbp!=1) && (vbcin!=1) && (vbgeo!=1))
{vbs=vbtbp*0.3+vbcin*0.4+vbgeo*0.3;           blancos(92);
  printf(texto,"%f",vbs); outtextxy(x4-42,y1+92,texto);vbsok=1;}
if ((vblm!=-1) && (vbtbp==1) && (vbcin!=1) && (vbgeo!=1))
{vbs=vblm*0.4+vbcin*0.3+vbgeo*0.3;           blancos(92);
  printf(texto,"%f",vbs); outtextxy(x4-42,y1+92,texto);vbsok=1;}
if (vbsok==0)      {vbs=vb;   if (vbs!=-1) {blancos(92);
  printf(texto,"%f",vbs); outtextxy(x4-42,y1+92,texto);}}
  setcolor(2);
}

```



```

compararesultadosda()
{
    BOOLEAN dasok;
    char texto[20];
    dasok=0;
    setcolor(22);
    if ((dalm!=-1) && (datbp==-1) && (dacin==-1) && (dageo==-1))
{das=dalm;    blancos(70);    sprintf(texto,"%f",das);    outtextxy(x4-
    42,y1+70,texto); dasok=1;}
    if ((dalm==-1) && (datbp!=-1) && (dacin==-1) && (dageo==-1))
{das=datbp; blancos(70);    sprintf(texto,"%f",das);    outtextxy(x4-
    42,y1+70,texto); dasok=1;}
    if ((dalm==-1) && (datbp==-1) && (dacin!=-1) && (dageo==-1))
{das=dacin; blancos(70);    sprintf(texto,"%f",das);    outtextxy(x4-
    42,y1+70,texto); dasok=1;}
    if ((dalm==-1) && (datbp==-1) && (dacin==-1) && (dageo!=-1))
{das=dageo; blancos(70);    sprintf(texto,"%f",das);    outtextxy(x4-
    42,y1+70,texto); dasok=1;}
    if ((dalm!=-1) && (datbp!=-1) && (dacin==-1) && (dageo==-1))
{das=dalm*0.5+datbp*0.5;    blancos(70);    sprintf(texto,"%f",das);
    outtextxy(x4-42,y1+70,texto); dasok=1;}
    if ((dalm!=-1) && (datbp==-1) && (dacin!=-1) && (dageo==-1))
        {das=dalm*0.5+dacin*0.5;    blancos(70);
    sprintf(texto,"%f",das); outtextxy(x4-42,y1+70,texto); dasok=1;}
    if ((dalm!=-1) && (datbp==-1) && (dacin==-1) && (dageo!=-1))
{das=dalm*0.5+dageo*0.5;    blancos(70);    sprintf(texto,"%f",das);
    outtextxy(x4-42,y1+70,texto); dasok=1;}
}

```

```

if ((dalm===-1) && (datbp!=-1) && (dacin!=-1) && (dageo===-1))
{ das=datbp*0.5+dacin*0.5;   blancos(70);   sprintf(texto,"%f",das);
  outtextxy(x4-42,y1+70,texto); dasok=1;}
if ((dalm===-1) && (datbp!=-1) && (dacin===-1) && (dageo!=-1))
{ das=datbp*0.5+dageo*0.5;   blancos(70);   sprintf(texto,"%f",das);
  outtextxy(x4-42,y1+70,texto); dasok=1;}
if ((dalm===-1) && (datbp===-1) && (dacin!=-1) && (dageo!=-1))
{ das=dacin*0.5+dageo*0.5;   blancos(70);   sprintf(texto,"%f",das);
  outtextxy(x4-42,y1+70,texto); dasok=1;}
if ((dalm!=-1) && (datbp!=-1) && (dacin!=-1) && (dageo===-1))
{ das=dalm*0.4+datbp*0.3+dacin*0.3;           blancos(70);
  sprintf(texto,"%f",das); outtextxy(x4-42,y1+70,texto); dasok=1;}
if ((dalm===-1) && (datbp!=-1) && (dacin!=-1) && (dageo!=-1))
{ das=datbp*0.3+dacin*0.4+dageo*0.3;           blancos(70);
  sprintf(texto,"%f",das); outtextxy(x4-42,y1+70,texto); dasok=1;}
if ((dalm!=-1) && (datbp===-1) && (dacin!=-1) && (dageo!=-1))
{ das=dalm*0.4+dacin*0.3+dageo*0.3;           blancos(70);
  sprintf(texto,"%f",das); outtextxy(x4-42,y1+70,texto); dasok=1;}
      if (dasok==0)   { das=da;   if (das!=-1) {blancos(70);
  sprintf(texto,"%f",das); outtextxy(x4-42,y1+70,texto);}}
setcolor(2);
}

```

CAPITULO VI

6. Conclusiones y recomendaciones

6.1. Conclusiones

1. La táctica submarina ha desarrollado a través de su historia, diferentes métodos y procedimientos de búsqueda, aproximación y ataque a buques blanco.
2. La gran restricción de utilizar los sensores en forma pasiva, logró el desarrollo de diversas fórmulas matemáticas y algoritmos para llegar a la solución del problema de tiro.
3. Es conveniente incorporar a la nueva tecnología informática, parte de los algoritmos matemáticos probados a través de décadas, los mismos que son desarrollados manualmente utilizando pizarras acrílicas, ábacos de material plástico y muchos cálculos mentales.
4. Un software proporcionaría soluciones en forma rápida y exacta, sirviendo de excelente soporte a las decisiones del comando durante las fases de aproximación y ataque submarino

6.2. Recomendaciones

1. La Marina de Guerra debe considerar la implementación de un “sistema de control de tiro para torpedos a bordo de submarinos” como medio alternativo al computador principal, el cual consolidaría gran parte de los ploteos utilizados manualmente por el Team de Ataque en la actualidad.

ANEXOS

Anexo 01: Estimación de costos.

	US\$
a. Gastos de Personal:	
- 1 analista x (30 horas semanales) x 14 semanas x US\$10.00 hora	4 200.00
- Asesoría	1 500.00
- Apoyo secretarial	300.00
b. Gastos de Equipo y Software	
- 500 horas de uso de PCs	1 000.00
- Licencias	400.00
c. Otros Gastos	740.00

TOTAL COSTO ESTIMADO DEL PROYECTO	
(Sin I.G.V)	US\$ 8 140.00

BIBLIOGRAFÍA

R KONG MAYNARD: “Lenguaje de Programación C”, Pontificia Universidad Católica del Perú Fondo Editorial 1989.

SCOTT ZIMMERMAN / BVERLY B. ZIMMERMAN: “La biblia del Turbo C”, Ediciones Anaya Multimedia S.A., 1990.

ESCUELA DE SUBMARINOS: “Manual de Táctica de la Fuerza de Submarinos” (PLOSUB-21019), Lima 1980

ESCUELA DE SUBMARINOS: “Manual de Táctica avanzada de la Escuela de Submarinos”, Lima 1957.

ESCUELA DE SUBMARINOS: “Método Autodidacta Aprendizaje Táctico de la Escuela de Submarinos”, Lima 1992.

REFERENCIAS ELECTRÓNICAS

GRUPO RADIO CENTRO **Historia de los Submarinos - 3 de junio de 2006**, [en línea]. México 2006.[citado 19 Junio 2006]. Microsoft HTML. Disponible en : [http:// ww.radiocentro.com/grc/redam.nsf/vwALL/XPAO-6QGP83](http://ww.radiocentro.com/grc/redam.nsf/vwALL/XPAO-6QGP83) - 35k -