

2
2 Gen.

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA



**COORDINACION VISUOMOTORA EN LA
MANTIS RELIGIOSA.
UN MODELO COMPUTACIONAL.**

T E S I S

QUE PARA OBTENER EL TITULO DE:
INGENIERO EN COMPUTACION
P R E S E N T A:
VICENTE ALCANTARA GUERRERO

DIRECTOR DE TESIS
DR. ROLANDO LARA Y ZAVALA



MEXICO. D. F.

1985



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

I N D I C E

INTRODUCCION.	1
A. Por qué se estudia a la mantis religiosa ?	1
B. Aspectos generales de la mantis religiosa.	4
C. Objetivos del trabajo de tesis.	8
D. Organización del trabajo de tesis.	11
E. Infraestructura utilizada.	12
I. TEORIA DE ESQUEMAS.	15
I.1. Teoría de esquemas. Definición.	15
I.2. Etograma de espacio de estados de la M.R.	17
I.2.1. Captura de presa.	18
I.2.2. Respuesta ante un depredador.	20
I.2.3. Objetos sin movimiento. Alcance de refugio.	23
II. MODELO TEORICO DE COMO LA M.R. BUSCA Y ALCANZA	
UN REFUGIO.	31
II.1. Introducción.	31
II.2. Experimentos realizados y obtención del modelo. ...	32
II.3. Implementación y simulación del modelo de alcance-	
de refugio.	36
II.3.1. Consideraciones y restricciones del	
programa de simulación.	38
II.3.2. Conformación del programa de simulación. ...	39
II.3.3. Forma de uso del programa de simulación. ...	46
III. SALTO BALISTICO. OBTENCION DE MODELO MATEMATICO.	65
III.1. Introducción.	65
III.2. Modelo Matemático del Salto Balístico.	66
III.3. Resultados.	69
IV. SIMULACION DE LA RESPUESTA DE ATAQUE HACIA UNA PRESA DE-	
LA MANTIS RELIGIOSA.	77
IV.1. Introducción.	77
IV.2. Modelo matemático.	78
IV.3. Simulación del modelo matemático.	82
V. CONCLUSIONES.	98
APENDICES.	
A. PROGRAMA DE SIMULACION DE ALCANCE DE REFUGIO.	101
B. PROGRAMA ANALISIS DE SALTO BALISTICO.	134
C. PROGRAMA DE SIMULACION DE ATAQUE.	150
D. BIBLIOTECAS DE SOFTWARE.	161
BIBLIOGRAFIA.	166

I N T R O D U C C I O N .

A. POR QUE SE ESTUDIA A LA MANTIS RELIGIOSA ?.

Uno de los problemas de mayor importancia en el estudio del sistema nervioso es el de analizar los mecanismos neuronales que controlan la conducta del animal: como la información sensorial se convierte en la coordinación de pautas motoras que le permiten al animal confrontar adecuadamente su situación presente. Para ello han surgido recientemente enfoques multidisciplinarios que buscan relacionar observaciones etológicas, como la respuesta de orientación hacia una presa, la respuesta de huida a un depredador, con los posibles mecanismos neuronales responsables de ellas. Estos estudios se complementan conceptualmente con modelos matemáticos y computacionales que permiten establecer hipótesis específicas, analizar cuantitativamente sus consecuencias, definir nuevos experimentos empíricos y guiar conceptualmente tanto la investigación etológica como la neurofisiológica al predecir y postular diversas funciones neuroetológicas. Para este tipo de estudios se han elegido animales cuya conducta pueda ser estudiada fácilmente y que su sistema nervioso no sea tan complicado como el de los vertebrados superiores. Los animales que se han elegido para estos estudios son generalmente vertebrados inferiores, como los anfibios, o invertebrados, como moluscos, crustáceos e insectos.

En el presente trabajo de Tesis estudiaremos algunos aspectos de la coordinación visuomotora en el insecto Mantis Religiosa (M.R.), tales como son las conductas: 1) alcance de refugio en un ambiente de objetos sin movimiento (llamada *chantlitaxia*), 2) el ataque de una presa, y 3) la respuesta de imitación ante un depredador. Para la primera conducta se llevarán a cabo estudios etológicos (experimentos de comportamiento), la obtención de un modelo teórico que defina esta conducta y su simulación a través de graficación animada tridimensional; para la segunda conducta existe un modelo teórico definido, haciéndose la proposición de un nuevo modelo basándose en el ya existente y llevando a cabo su simulación gráfica animada en dos dimensiones en computadora digital; la tercera conducta ya ha sido estudiada anteriormente por investigadores diversos, tratándose en este trabajo su definición formal y su inclusión en un esquema funcional motor que define a las tres conductas anteriormente mencionadas (además de la de apareamiento ante una pareja) en una forma integral. Todos estos estudios forman parte de un importante proyecto de trabajo del Laboratorio de Bioingeniería del Centro de Investigaciones en Fisiología Celular de la U.N.A.M.

La selección de este animal se debe a que presenta grandes ventajas para este tipo de estudios, entre las que podemos mencionar las siguientes: su conducta está regulada básicamente por la visión; sus respuestas a estímulos visuales son muy estereotipadas por lo que pueden ser

fácilmente estudiadas; son animales casi estáticos; y su sistema nervioso no es tan complejo como el de los vertebrados. Hasta el momento estos animales sólo han sido estudiados desde un punto de vista etológico, por lo cual nos limitaremos a describir en términos funcionales las operaciones que tiene que realizar su sistema nervioso durante las tres conductas anteriormente mencionadas, dejando para investigaciones futuras el análisis de los posibles mecanismos neuronales responsables de ellas.

B. ASPECTOS GENERALES DE LA MANTIS RELIGIOSA.

A continuación se citan algunos aspectos generales y estudios sobre la Mantis Religiosa.

TAXONOMIA DE LA MANTIS RELIGIOSA.

Clase:	Insecta o Hexápoda.
Orden:	Orthoptera.
Suborden:	Dictyoptera.
Superfamilia:	Mantoidea.
Familia:	Mantidae.
Género:	Mantis(1800 especies).
Especie:	Mantis Religiosa.

ESTUDIOS ETOLOGICOS DE LA COORDINACION VISUOMOTORA.

1.- Estos animales atacan a estímulos pequeños de diversas formas, integrando información sobre su tamaño, alas, movimiento, etc. [04]. Un estudio sistemático de cómo se modifica la respuesta de orientación y de ataque a diferentes configuraciones del estímulo no se ha realizado.

2.- La respuesta de ataque del animal está controlada por una interacción entre el sistema visual y un grupo de mecanoreceptores que definen la posición de la cabeza con respecto al cuerpo del animal [05].

3.- Estos animales presentan una respuesta

característica, llamada respuesta deimática, ante la presencia de un predador [02]. Los predadores que se han usado para desencadenar estas respuestas son diversos tipos de aves. Sin embargo, un estudio sistemático de la configuración típica de un estímulo tipo predador no se ha realizado.

4.- La profundidad de objetos en movimiento es estimada por el animal a través de mecanismos binoculares [02,03].

5.- Estos animales aprenden a no atacar a estímulos que después de varios intentos no pueden atrapar [08].

6.- La respuesta deimática se habitúa ante la presentación repetitiva del estímulo [01]. La habituación de la respuesta de orientación no se ha estudiado sistemáticamente en estos animales.

ESTUDIOS ANATOMICOS.

1.- El sistema nervioso de estos animales esta dividido en varios ganglios. Figura F.1. a) El cerebro o ganglio supraesofágico; b) el ganglio subesofágico; c) 3 ganglios torácicos; y d) 6 ganglios abdominales [05].

2.- El cerebro se divide en tres zonas principales: proto, deuto y trito cerebro. El protocerebro recibe fibras de neuronas de vías visuales así como de los ocelos, conteniendo dos grandes poblaciones neuronales que se ordenan especial y simétricamente formando los cuerpos pedunculados y una estructura altamente ordenada denominada

cuerpo central complejo. El deutocerebro recibe fibras sensoriales de numerosos sinsila localizadas en las antenas y contiene grupos de neuronas motoras que inervan los músculos antenales. Las neuronas en el tritocerebro conectan el cerebro con la parte anterior del sistema estomatogástrico y además unen las partes dorsales del cerebro con la cuerda nerviosa central y el ganglio subesofágico [05].

4.- El complejo subesofágico a su vez esta compuesto por tres pares de ganglios que inervan los órganos sensoriales y los músculos de las zonas bucales, glándulas salivales, algunos músculos y receptores del cuello [05]

5.-El sistema nervioso torácico o cuerda nerviosa ventral incluye nervios pro- meso y meta torácicos que en ocasiones se encuentran fusionados. Cada ganglio torácico representa el centro sensorial y motor del segmento respectivo, aun cuando algunos de los músculos que intervienen en la locomoción reciben aferentes de neuronas motoras de ganglios de otros segmentos [05].

ESTUDIOS POR MEDIO DE LESIONES EN LOS GANGLIOS

1.- Al destruirse el cerebro, el animal camina indiscriminadamente hacia adelante siendo incapaz de interactuar con los objetos fijos de su universo [05].

2.- Al destruirse el núcleo subesofágico, el animal queda totalmente inmobilizado [05].

3.- Al destruirse parcialmente la zona lateral del cerebro, el animal tiende a rotar continuamente hacia el lado de la lesión [05].

4.- Al dividirse al cerebro en dos partes, el animal queda totalmente inmovilizado, aunque se observa que el animal tiende a realizar movimientos motores, pero es incapaz de ejecutarlos [05]. Todo esto sugiere que el cerebro esta inhibiendo continuamente al ganglio subesofágico que se encarga de activar a los ganglios torácicos. Asimismo, parece que entre el lóbulo izquierdo y el derecho del cerebro existe una inhibición mutua, que controla el movimiento de rotación hacia uno u otro lado del animal. Es necesario, sin embargo, hacer un estudio más riguroso de cómo estas zonas cerebrales controlan las múltiples respuestas del animal tanto para estímulos en movimiento como para objetos fijos.

C. OBJETIVOS DEL TRABAJO DE TESIS.

La M.R. es un animal que permanece estático al cubierto de un refugio y se ha observado que 4 de sus pautas motoras muy importantes que realiza son las siguientes:

- a) Apareamiento ante una pareja.
- b) Captura de una presa.
- c) Respuesta de repelencia ante un depredador.
- d) Búsqueda de un refugio cuando está al descubierto.

Estas conductas tienen la propiedad de ser perfectamente estereotipadas, haciéndose notar que:

-La conducta (a) no ha sido estudiada estrictamente.

-Las conductas (b) y (c) han sido estudiadas extensamente e incluso la conducta (b) esta definida por un modelo de control[05].

-La respuesta (d) es una conducta que no ha sido estudiada debido a que ésta fue reconocida como tal en el Centro de Investigaciones en Fisiología Celular, y que contempla su estudio una variedad de tácticas que utiliza la M.R. en el alcance de un refugio como son percepción de profundidad, subir y bajar obstáculos, bajar y saltar zanjás, y en general todo lo concerniente a la percepción de un mundo tridimensional simplificado de este insecto.

En la tarea de profundizar en el conocimiento del sistema nervioso de la M.R. este trabajo se centra sólo en dos de los 4 tipos de comportamiento antes mencionados y éstos son el alcance de un refugio y la captura de una presa.

En la primera se lleva a cabo la obtención de los modelos matemáticos y su simulación en computadora para definir la conducta de la M.R. para alcanzar un refugio. Cuando la M.R. se encuentra al descubierto, lo primero que hace es tratar de encontrar un sitio seguro, para de esta forma no ser descubierta por una presa o un depredador. Se ha observado que le es atractivo un lugar vertical y elevado, y que salva todo tipo de obstáculos y situaciones para llegar a él. Si al tratar de alcanzar un refugio encuentra a su paso diferentes obstáculos, la M.R. lleva a cabo diferentes acciones para salvarlos de acuerdo a la localización y dimensión de éstos.

En la segunda se lleva a cabo la formalización matemática y la simulación del modelo de control propuesto por H. Mittelstaedt en la captura de una presa por la M.R.

Los resultados de la simulación se mostrarán gráficamente con técnicas de animación. Esto nos permitirá no solo tener un reporte cuantitativo del comportamiento simulado por medio de los modelos, si no que además nos permitirá seguir visualmente la evolución de la conducta de la M.R. ante una situación dada en su medio ambiente tal como si estuviéramos observando el fenómeno realmente, dándonos la capacidad de hacer un análisis cualitativo.

Los resultados obtenidos en este trabajo son obviamente de interés para algunas disciplinas Biológicas, pero además es de suma importancia para disciplinas de Ingeniería y en especial para la Robótica, una ciencia que en este momento esta siendo impulsada grandemente debido a la infinidad de

aplicaciones que tiene. El hecho de que la conformación del sistema nervioso de la M.R. sea muy elemental y realice funciones de cierta complejidad sugiere una factibilidad de implementación en robots de un sistema funcional similar al sistema nervioso de la M.R. para que los modelos matemáticos etológicos, objeto de nuestro estudio, sean aplicados en el desarrollo de esquemas y técnicas visuomotoras para robots como programación de movimientos, percepción de su espacio tridimensional, percepción de profundidad de objetos, que, precisamente debido a su simplicidad, involucran un abatimiento de costos no sólo monetarios sino de proceso, que es en este momento, al igual que la parte mecánica, la parte crítica de ésta rama de la tecnología.

D. ORGANIZACION DEL TRABAJO DE TESIS.

La forma en que se presentará este trabajo es la siguiente.

En el punto E de esta introducción se cita la infraestructura utilizada en este trabajo

En el capítulo I se habla de lo que es la teoría de esquemas y se definen, explican y discuten los etogramas de la M.R. en función de esta teoría.

En el capítulo II se trata la obtención del modelo matemático y su simulación del comportamiento de la búsqueda de un refugio en un espacio tridimensional estático y de un número de objetos con características limitadas.

En el capítulo III se describen los experimentos y técnicas utilizadas para definir el modelo matemático del llamado salto balístico, el cual es una táctica utilizada por la M.R. en el alcance de un refugio y que por ser de suma importancia se presenta en forma separada de la obtención del modelo teórico del alcance de un refugio, tratado en el capítulo II.

El capítulo IV trata lo concerniente al desarrollo matemático y simulación del modelo de control del ataque de una presa.

Finalmente en el capítulo V se dan conclusiones y observaciones del trabajo de tesis.

E. INFRAESTRUCTURA UTILIZADA.

EL equipo con que se contó para llevar a cabo este trabajo de tesis es el siguiente:

- Microcomputadora Cromemco System III.

Características de Hardware.

Microprocesador: Z80-A 4 Mhz.
Memoria RAM: 64K bytes
Sistema de Bus: S-100 standard
Memoria Secundaria: 2 Disk drives DD-DS; 8"
con capacidad de 1 Mbyte cada uno.
Impresora: 80 columnas; 150 cps.
Terminal 'Inteligente': 80 cpl-24 líneas.
Interface Gráfica SDI.

Alta resolución de 754 X 482 pixels
a color.
2 Páginas de memoria de 48k
para almacenar una Gráfica completa
en cada una permitiendo efectos
como animación y ventanas.

Monitor RGB 13"
Color.

Características de Software.

Lenguajes: Ratfor, Fortran y macro-assembler Z80.

Bibliotecas: Matemáticas de Fortran.

Gráficas de SDI. Compuesta por
rutinas que llevan a cabo
funciones como:
a) Escalar espacio de graficación.
b) Pintar una línea.
c) Pintar un punto.
d) Definir color de graficación.
e) Selecc. página de graficación.
f) Selecc. página de trabajo.
g) Borrar página de graficación.
h) Borrar página de trabajo.

- Videocassettera. Sony 5800.

Formato: Beta.

- Televisión y cámara de video de color.

- Sistema electrónico de procesamiento de imagen.

Sistema que lleva a cabo la cuantificación en tiempo real de las coordenadas de un punto luminoso [mantis religiosa tratada con pintura] en movimiento tomado de la imagen de la cámara de video produciendo un archivo de datos en la computadora, teniéndose con esto la evolución de un movimiento realizado por la mantis religiosa que toma sólo una fracción de segundo en llevarse a cabo para su posterior análisis.

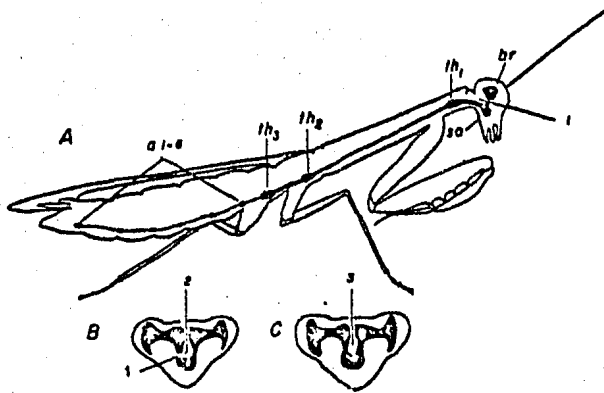


Figura F.1. (A) Diagrama del sistema nervioso central de la mantis religiosa. a1-a6, ganglios abdominales; hr, cerebro; so, ganglio subesofágico; th1-th3, ganglio torácico. (B, C) vistas anteriores del cerebro y ganglio subesofágico. 1, corte para inhibir al cerebro; 2, división del cerebro por la mitad; 3, separación del cerebro.

C A P I T U L O I
T E O R I A D E E S Q U E M A S .

I.1 TEORIA DE ESQUEMAS. DEFINICION.

La teoría de esquemas propone que el Sistema Nervioso de los animales está organizado por estructuras de información que, siguiendo a Piaget[10], denominamos esquemas, los cuales permiten al animal definir la pauta motora adecuada para alcanzar un objetivo dado. La teoría de esquemas es una teoría general de las estructuras de información que pueden utilizar animales y robots para poder definir su respuesta motora ante una circunstancia específica. Un esquema se define como una estructura de información que relaciona una circunstancia visual específica con una serie de conductas motoras [06].

Podemos hacer una analogía de que los esquemas son para animales y robots lo que los diagramas de flujo para los programas de las computadoras digitales.

Un esquema tiene siempre una meta a la cual busca llegar a través de las diferentes acciones motoras. El esquema esta constituido por los siguientes elementos:

1) Esquemas perceptuales que reconocen objetos significativos en el animal o robot. En el caso del animal puede tratarse de una presa, un depredador, una barrera etc.; en el caso de un robot puede ser una pieza para ensamblar, un switch, un objeto para transportar etc. Asimismo los esquemas perceptuales definen la métrica del

espacio externo, es decir, la distancia de la presa, la altura del switch, etc.

2) Esquemas motores que representan las pautas motoras fundamentales del animal o del robot, como son caminar, orientar, brincar etc.

3) Programa de coordinación sensorimotora que define la secuencia de activación de esquemas perceptuales y motores para alcanzar un fin específico.

4) Existe competencia entre diversos esquemas del mismo nivel, es decir una relación heterárquica, para controlar la conducta motora.

5) Existe una cooperatividad entre los diferentes esquemas para alcanzar el estado meta postulándose una relación jerárquica donde el esquema principal define la meta y llama a un conjunto de esquemas, cada uno con una submeta específica, para alcanzar el objetivo.

6) Existe un estado meta al cual tiende la actividad concertada de un grupo de esquemas sensoriales y motores.

7) Existen estados fallidos que representan que el esquema principal no pudo llegar a su objetivo.

8) El conjunto de esquemas que define la conducta del animal ante diferentes circunstancias se le llama etograma de espacio o esquema global del animal o robot.

Con la teoría de esquemas es posible explicar, a un nivel conceptual, la conducta visuomotora de la M.R.

I.2. ETOGRAMA DE ESPACIO DE ESTADOS DE LA M.R.

El etograma de espacio de estados general de la M.R. es mostrado en la figura F.I.1. Se observa que existen cuatro diferentes esquemas principales. 1).-en presencia de una presa; 2).-en presencia de un depredador; 3).-en presencia de una pareja para apareamiento; 4).-en presencia de un medio ambiente con estímulos sin movimiento. La conducta de apareamiento está incluida en el etograma general, pero ésta no es definida en este trabajo, ya que no existen estudios y elementos necesarios para hacerlo.

Los estímulos en movimiento están modulados por un sistema jerárquico que define cuál de ellos va a ser activado dependiendo de su experiencia. Estos tres esquemas iniciales están en competencia entre sí para tomar control de la conducta del animal; esta competencia, como ya se mencionó antes, está dirigida por el esquema general de estímulos en movimiento. Si este esquema no está activo, entonces el esquema de una circunstancia sin objetos móviles se activa.

Al activarse cualquiera de los esquemas principales, se define inmediatamente el objetivo o meta del mismo y se empiezan a activar diversos esquemas para alcanzarlo.

A continuación se dará lugar a la explicación de tres de los cuatro esquemas principales definidos en el etograma de espacio de estados general.

I.2.1. Captura de Presa.

En la figura F.I.2 se muestra el esquema principal Captura-de-presa.

Una vez que este esquema principal há sido activado, se observa que la primera acción motora a llevar a cabo es orientarse a la presa; en estas condiciones 2 posibles esquemas empiezan a competir entre sí para tomar control de la conducta del animal: 1)Presa-lejana y 2)Presa-al-alcance. Una vez que se activa uno de ellos, dependiendo de la condición métrica del espacio, es decir, en este caso de la distancia presa-mantis, se activan diferentes esquemas que a su vez pueden competir entre sí para tomar el control de la conducta.

En caso de activarse el esquema Presa-lejana, esto es, si la distancia presa-mantis es grande, compiten dos esquemas: 1a)Espera y 1b)Aproximate, de los cuales su activación estará regulada por el estado de hambre del insecto y tal vez por la presencia de obstáculos entre presa y mantis. En caso de ser activado el esquema Espera se provocará una acción de espera, para después activar el esquema Captura-de-presa, si durante el tiempo en que se llevo a cabo la ejecución del esquema Espera la presa se acercó a la mantis religiosa. En caso de que el esquema Aproximate sea activado, se lleva a cabo una acción motora de aproximación de la M.R. hacia la presa, para después

activar el esquema Captura-de-Presa, haciéndose una nueva evaluación de la métrica, pero ahora con la nueva posición de la M.R., producto del aproximamiento.

Si la distancia a la presa es pequeña, se activa el esquema Presa-al-alcance, la cual provocará una acción motora de ataque, llegando de esta forma al estado meta y en consecuencia al final de la ejecución de este esquema principal, pasando posteriormente al estado inicial A.

Como se observa, el esquema Captura-de-presa estará siendo activado por varias vías, y sólo termina hasta que se active el esquema Ataque, esto es, cuando la distancia presa-mantis es mínima, llegando de esta forma al estado meta y por tanto a su finalización

Para explicar esta conducta existe un sistema de control, el cual postula que los mecanismos involucrados antes de capturar una presa no son exclusivamente visuales, ya que se ha comprobado que existe un sistema que actúa conjuntamente con éste, y que ha sido llamado propioceptivo. En este sistema propioceptivo los receptores son vellocidades en el cuello de la M.R., los cuales, al ser presionados por un giro de la cabeza hacia un lado, generan impulsos nerviosos que son utilizados como complemento de la información visual, obteniendo de esta forma una posición exacta del objeto, dando como resultado un alto porcentaje de éxito en la captura de una presa. La simulación de esta conducta se llevará a cabo en base a este modelo de control, y que será tratado con mayor amplitud y profundidad en el capítulo IV.

I.2.2. Respuesta ante un depredador.

Antes de explicar el etograma correspondiente a la conducta Respuesta ante un depredador se darán algunos aspectos, considerados importantes, que forman parte de ésta. Se ha observado y estudiado que el insecto M.R. ante la presencia de un depredador, presenta una conducta llamada respuesta Deimática [02]. Esta conducta se refiere a una exhibición conspicua que se desarrolla cuando se enfrenta a un estímulo tipo depredador. Esta exhibición consta de varios componentes que se presentan en orden sucesivo y cuyo número depende de la peligrosidad que represente el estímulo tipo depredador (tamaño y distancia depredador-mantis); estas componentes son: 1) antenas dirigidas hacia atrás y divergentes; 2) sistema bucal abierto, mostrando las mandíbulas; 3) protórax levantado: primer par de patas flexionadas y lateralmente extendidas; 4) la tegmina se mantiene erecta con las superficies dorsales mirando hacia el depredador de tal manera que el estigma tegminario queda descubierto manteniendo las alas levantadas; 5) el abdomen se dobla hacia un lado mostrando una serie de bandas de color que de otra manera no son visibles; 6) estriludación que se logra frotando el abdomen hacia arriba y hacia abajo contra las alas; y 7) el insecto se mueve oblicuamente meciéndose violentamente de un lado a otro.

En el laboratorio del C.I.F.C. se ha observado que ante un estímulo tipo depredador la M.R. en ciertas condiciones

no presenta la respuesta deimática, haciéndose presente otro tipo de respuesta totalmente estereotipada, y que tentativamente se le llamó catalepsia simulada. Cuando la M.R. percibe el estímulo tipo depredador, ésta se queda totalmente estática, y por medio de movimientos pulsátiles, casi imperceptibles, busca adoptar una posición paralela respecto al piso o a un objeto vertical. La conducta es tan dramática que se ha observado que para una M.R. estando sobre un objeto a una altura de 4 cm. al detectar el estímulo, tarda aproximadamente 5 minutos en llegar al piso (el tamaño del cuerpo de la M.R. en cuestión es de 6 cm.), para quedar tirada sobre él, dando la apariencia de estar muerta.

En algunas ocasiones la M.R. huye o ataca al depredador, pero hasta el momento no se ha estudiado cuales son las condiciones necesarias para que se active alguno de las cuatro tipos de conducta, anteriormente mencionados, ante un depredador. Por lo anterior, el esquema de la respuesta ante un depredador, el cual es mostrado en la figura F.I.3, esta definido en términos cualitativos, el cual es significativo por representar un avance en el conocimiento de la conducta de los insectos, amén de incluir una respuesta recién encontrada como es la de catalepsia simulada.

Una vez que el esquema Respuesta-ante-depredador es activado, compiten dos esquemas por tomar la conducta. 1) Respuesta-deimática y 2) Respuestas-alternas, los cuales

serán activados de acuerdo con la fuerza del estímulo, en este caso el tamaño y la distancia presa-depredador.

Si la distancia es pequeña, se activa el esquema Respuestas-alternas, provocando que entren en competencia tres esquemas que son: 2a) Catalepsia-simulada; 2b) Ataca-a-depredador y 2c) Huida. Como se dijo, no están definidas aún las condiciones métricas para la activación de alguno de estos tres esquemas. Si se activa el esquema de Catalepsia-simulada, se lleva a cabo la conducta catalepsia simulada con todas sus componentes, para luego ir al estado inicial A. Si el esquema activado es el de Ataca-a-depredador, provoca una acción motora de ataque para después ir al estado inicial A. Finalmente, si se activa el esquema de Huida, se provoca una acción motora de huir, y se va al estado inicial A.

Si la distancia es mayor, se activa el esquema Respuesta-deimática, que provocará la acción motora de uno o más componentes de la respuesta deimática, estando este número de componentes en función de el tamaño del depredador y la distancia depredador-mantis; una vez presentado el número de componentes de la respuesta, correspondientes al estímulo, se va al estado inicial A.

Cabe hacer notar aquí que no está definido en forma explícita un estado meta, pero como dijimos, lo que trata de provocar la mantis es ahuyentar al depredador; de esta forma, el estado meta puede ser considerado cuando al regresar al estado inicial A ya no se reconoce al estímulo depredador, lo cual indica que se ha alejado.

I.2.3. Objetos sin movimiento. Alcance de refugio.

EL esquema correspondiente Objetos-sin-movimiento es presentado en la figura F.I.4. Cuando se presentan estímulos sin movimiento se activa este esquema, ejecutándose acciones motoras de orientación hacia refugio, balanceo (táctica en estudio en el C.I.F.C., pensándose que es utilizada por la M.R. como método de medición de profundidad para objetos sin movimiento) y de caminar. Después de realizar estas acciones, existe competencia de los esquemas: 1)Chantlitaxia y 2)No-chantli.

En caso de no existir refugio se activa el esquema No-chantli, el cual provoca una acción motora de caminar y se va al estado inicial A del etograma general.

En caso de que exista un refugio, se activa el esquema Chantlitaxia (atracción hacia un escondite), provocándose una competencia por tomar el control de la conducta entre dos esquemas, cuya activación dependerá de si existe o no obstáculo entre la mantis y el refugio; dichos esquemas son: a)No-obstáculo-mantis-chantli y b)Obstáculo-mantis-chantli.

Si el esquema No-obstáculo-mantis-chantli es el activado, entran en competencia los esquemas: 1a1)Aproximate-a-chantli y 1a2)Sube-chantli, cuya activación está en función de la distancia mantis-chantli; si la distancia es mínima, se ejecuta la acción motora sube chantli y alcanza el estado meta, para después pasar al

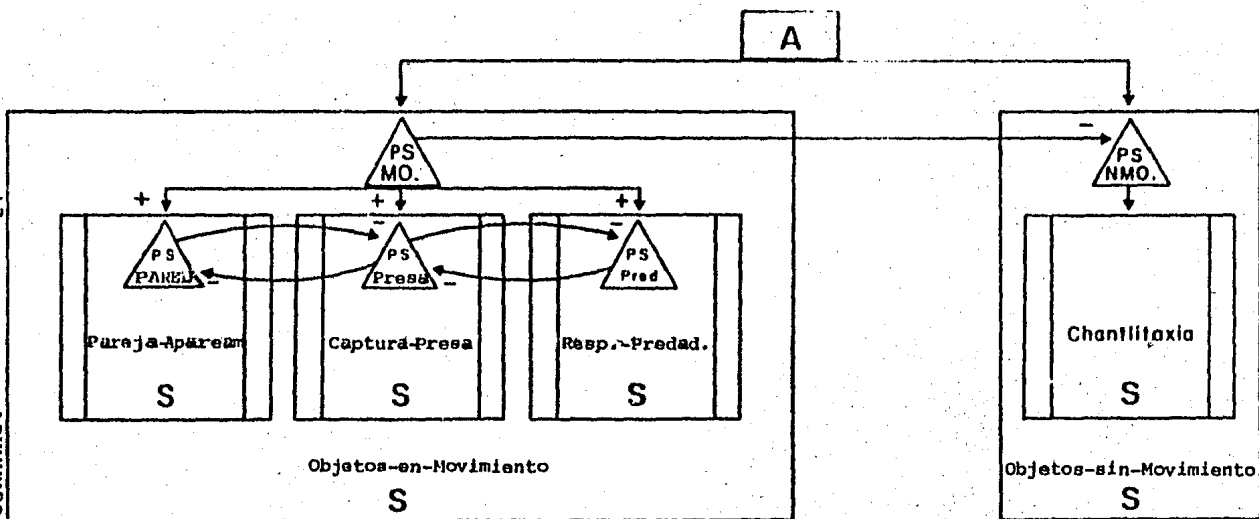
estado inicial A; si el esquema activado es Aproximate-a-chantli, es decir, la distancia es grande, se provoca una acción motora de aproximación y la invocación del esquema Objetos-sin-movimiento.

En caso de existir obstáculo entre la mantis y chantli, se activa el esquema Obstáculo-mantis-chantli, provocándose una competencia entre los esquemas: 1b1)Barrera y 1b2)Zanja; si el esquema activado es Barrera, en estas condiciones entran en competencia los esquemas: 1ba1)Barrera-objeto y 1ba2)Barrera-chantli, dependiendo su activación si la barrera es un objeto cualquiera o si es el chantli; en caso de activarse el esquema Barrera-objeto se provoca la realización de las acciones motoras de aproximate a barrera y sube barrera, invocándose posteriormente el esquema Objetos-sin-movimiento; en caso de que el esquema activado sea Barrera-chantli, se realizan las acciones motoras de aproximate a chantli y sube chantli, lográndose el estado meta, para después pasar al estado inicial A. En caso de ser activado el esquema Zanja, existe ahora la competencia de los esquemas: 1b2a)Baja; 1b2b)Pasar; 1b2c)Salto-balístico; 1b2d)Rodeo, cuya activación está en función de la profundidad y la longitud o distancia de la zanja, y en el caso de Pasar, también de la altura del objeto que se encuentra al frente. Si se activa el esquema Baja, consecuencia de profundidad mínima y distancia mayor, se provoca la acción motora de bajar la zanja; en la activación del esquema Pasar, producto de distancia mínima y profundidad del otro extremo de la zanja mayor o igual sobre

el que está, se provoca la competencia de dos esquemas: 1b2b1)Pasa-a-obstáculo y 1b2b2)Pasa-a-chantli, los cuales serán activados si el otro extremo de la zanja al que va a pasar es el chantli o u otro objeto. Si se activa Pasa-a-obstáculo, implica la acción motora de pasar a otro obstáculo por la simple extensión de las patas delanteras y se pasa al esquema Objetos-sin-movimiento. Si el esquema activado en Pasar es Pasa-a-chantli, se realiza la acción motora de pasar a chantli y trepa hasta el tope de chantli, lográndose el estado meta y pasándose al estado inicial A. Si el esquema activado en Zanja es salto-balístico, debido a condiciones de profundidad y distancia grandes, entran en competencia los esquemas de: 1b2c1)Balístico-obstáculo y 1b2c2)Balístico-chantli, que dependerá su activación si el salto es realizado hacia un obstáculo o hacia chantli. Si se activa el esquema Balístico-obstáculo, se realiza la táctica de salto balístico(explicada total y detalladamente en el Capítulo III), y se va a activar el esquema Objetos-sin-movimiento. Si el objeto que está al frente es chantli, se activa Balístico-chantli, que realiza la táctica de salto balístico hacia chantli, sube hasta el tope de chantli, y se cumple con el estado meta, pasándose posteriormente al estado inicial A; si la distancia y la profundidad son muy grandes se activa el esquema Rodeo, el cual provoca la ejecución del esquema perceptual de buscar ruta alternativa, orientarse y caminar, para después activar nuevamente al esquema Objetos-sin-movimiento.

Hasta aquí se tienen definidos los mecanismos visuomotores de las conductas más relevantes de la mantis religiosa y de las cuales hay elementos suficientes para desarrollar modelos que traten de definir estas conductas.

En los capítulos II y IV se presenta la obtención de los modelos matemáticos y su simulación gráfica animada de 2 conductas: en presencia de una presa y en presencia de un medio ambiente con estímulos estáticos o sin movimiento.



ESQUEMA PRINCIPAL (S)



ESQUEMA PERCEPTUAL (PS)

Figura P.1.1. Etograma de Espacios de Estados general de la Mantis Religiosa.

ESQUEMA CAPTURA - PRESA

- ⬡ Estado (GS)
Meta
- ⬠ Esquema (PS)
Perceptual
- Acción (MS)
Motora
- ▭ Esquema (S)

CAP I TEORIA DE ESQUEMAS. ETOGRAMAS.

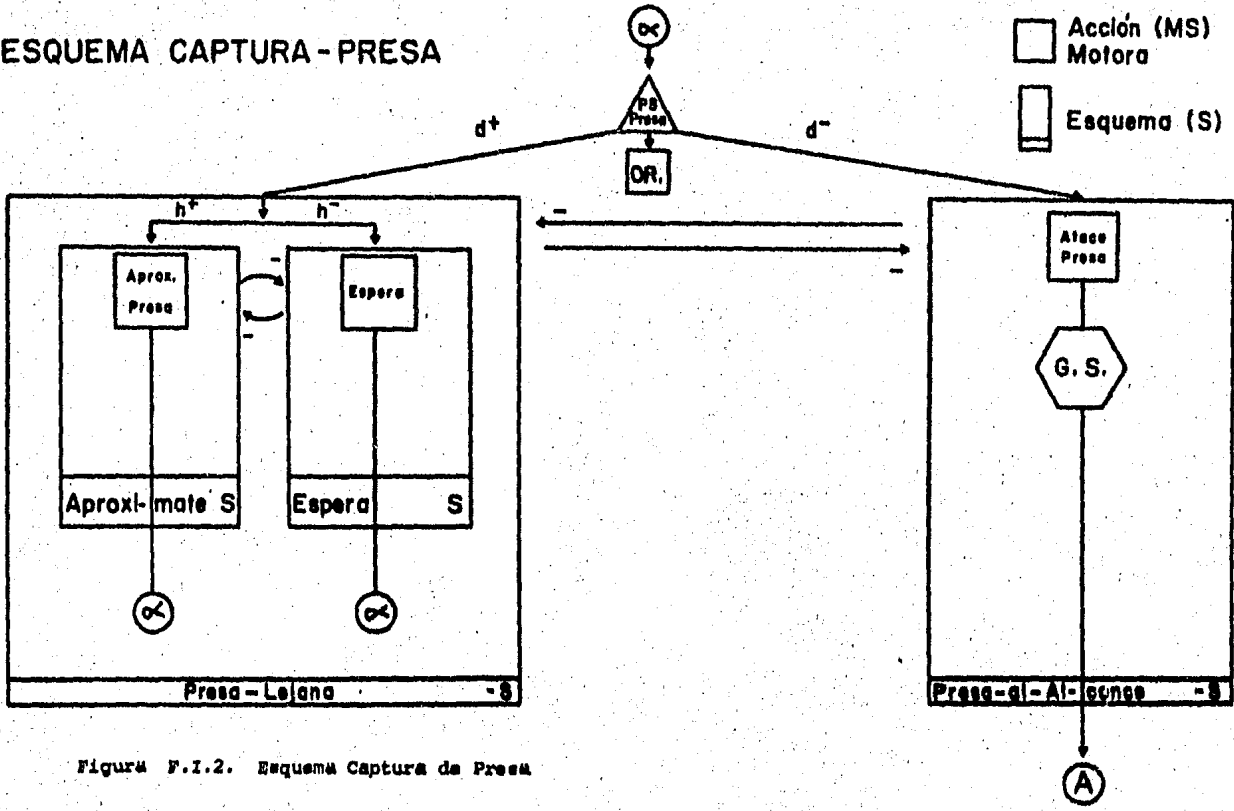


Figura F.1.2. Esquema Captura de Presca

Esquema Respuesta ante Depredador

29

CAP I TEORIA DE ESQUEMAS: ETOGRAMAS.

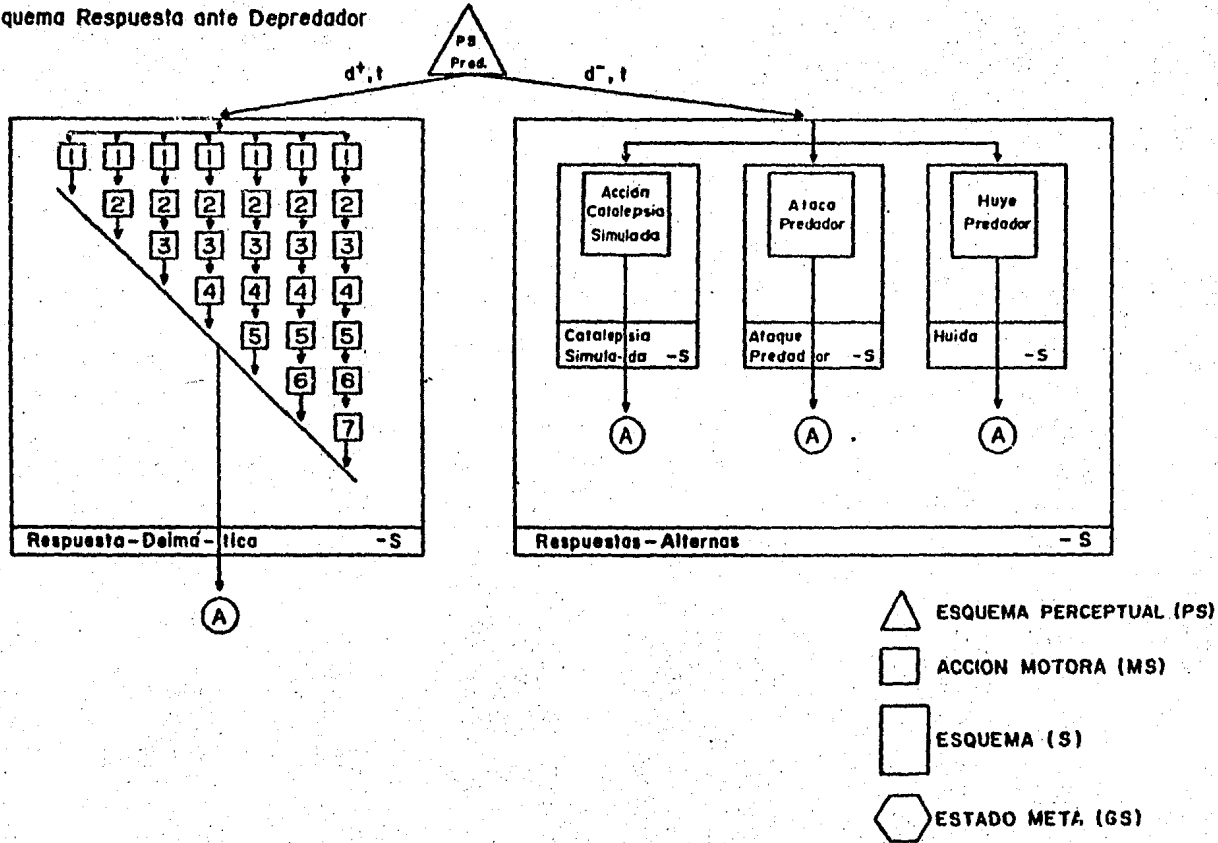


Figura F.I.3. Esquema Respuesta ante Depredador.

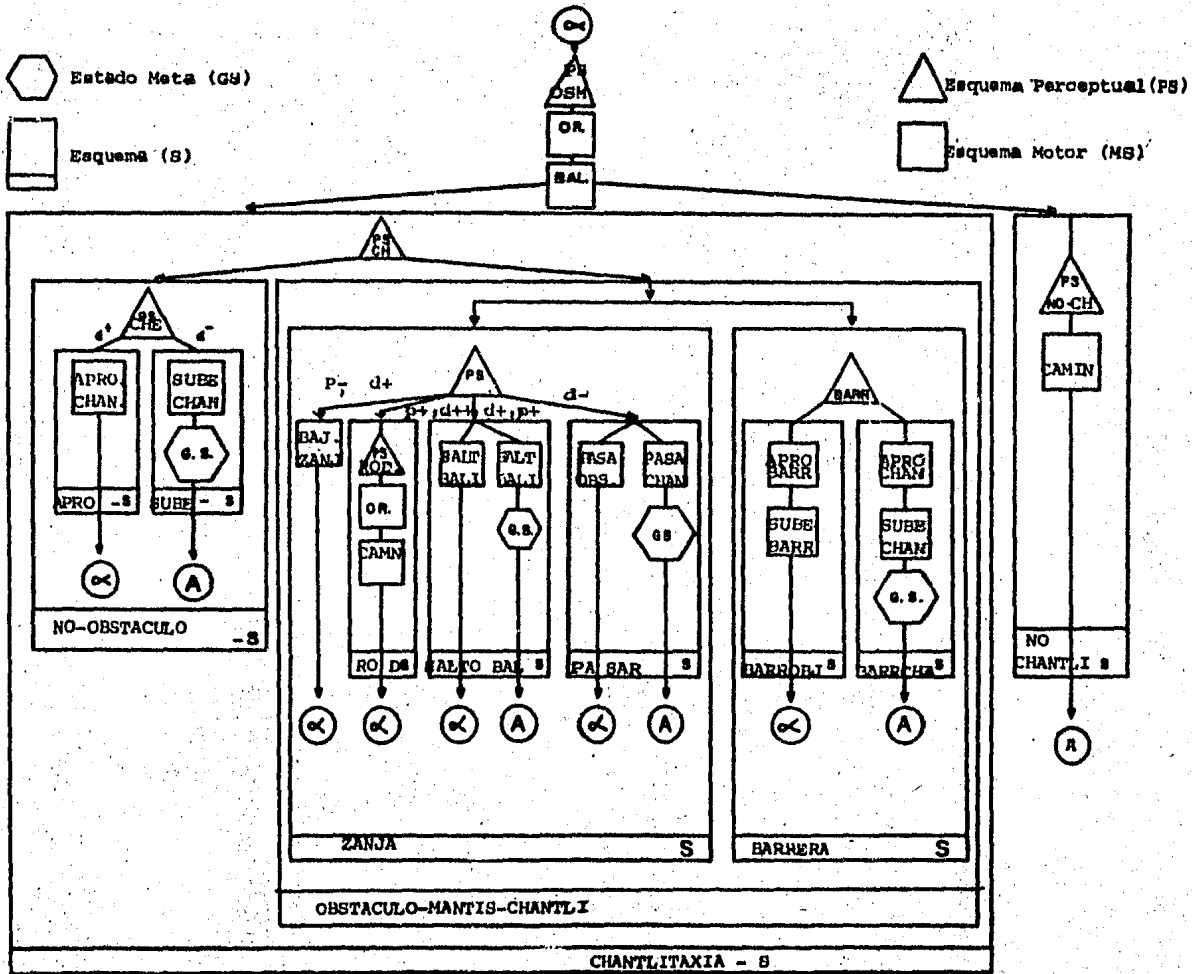


Figura F.I.4 Esquema Objetos- sin- Movimiento.

CAPITULO II

MODELO TEORICO DE COMO LA M R.
BUSCA Y ALCANZA UN REFUGIO.

II.1. INTRODUCCION.

Debido a que no existen estudios de como la mantis religiosa(M.R.) percibe objetos fijos en su mundo tridimensional y cómo en base a esa percepción define su conducta motora, en este capítulo se analizó una conducta muy estereotipada en estos animales, por medio de la cual buscan ubicarse en un lugar que les permita pasar desapercibidos tanto para depredadores como de posibles presas. En forma tentativa hemos llamado a esta conducta Chantli-taxia (Chantli quiere decir escondite en Náhuatl y taxis es una palabra del latín que se puede traducir como fuerza de atracción. De hecho parece que los antiguos mexicanos llamaban a la mantis religiosa campamocha que en Náhuatl quiere decir : donde está tu escondite "Campa mo chantli").

Esta conducta comprende los siguientes elementos:
1)busqueda del posible escondite-- cuya propiedad más importante es que sea un lugar alto-- por medio de una conducta de balanceo de su cuerpo, que probablemente es una táctica de la mantis religiosa para percibir a qué distancia se encuentran los objetos fijos; 2)orientación hacia el escondite; 3)aproximación al escondite; en este punto puede

haber uno o más obstáculos entre el animal y el escondite para los cuales el animal presenta varias conductas específicas en cada obstáculo; 4) alcanzar la parte superior del refugio.

Para poder estudiar esta conducta se diseñó una serie de experimentos con una variedad de características y situaciones, y se desarrolló un modelo teórico como resultado del análisis de los datos obtenidos. Asimismo, se realizó la simulación en computadora digital del modelo desplegado los resultados por medio de graficación tridimensional animada.

A continuación se citan los métodos y técnicas utilizados tanto para análisis, como para el desarrollo del modelo de esta conducta.

II.2 EXPERIMENTOS REALIZADOS Y OBTENCION DEL MODELO.

Una vez que se reconoció esta conducta como tal, se presentaron las siguientes interrogantes:

1) Cuáles son las características del objeto utilizado como refugio?.

2) Existe alguna regla en la definición de la trayectoria hacia el refugio?.

3)Cuál es la influencia de la existencia de variados objetos en su espacio tridimensional?.

La evolución de los experimentos en la búsqueda de las

respuestas a las cuestiones anteriores es como sigue. La mayor parte de ellos son preliminares, pero los resultados y el análisis del modelo desarrollado servirán para definirlos con más precisión. Para poder definir cuáles eran las características del refugio se puso al animal en un sitio al descubierto y se practicaron las siguientes situaciones.

1) se le colocaron uno a uno objetos de diferentes dimensiones en repetidas ocasiones, observándose que se dirigía hacia cada uno de los objetos que le fueron presentados y los trepaba. Posteriormente, los objetos fueron colocados en diferentes combinaciones de número y medidas frente a la mantis, notándose, al realizar una inspección de los resultados, que el lugar que prefería la mantis como escondite era el más alto, concluyéndose que precisamente el lugar escogido como escondite debe ser el más alto de los que existen en su espacio tridimensional. Todavía se necesita estudiar si el animal hace discriminación de otras características como grosor, color etc.

Después de múltiples observaciones de la respuesta de la M.R. ante un sólo objetivo, se observó que el animal sigue una trayectoria rectilínea, tanto si había obstáculos o no.

2) se colocaron de uno a cuatro objetos con diferentes combinaciones de tamaño y en una disposición que uno o más objetos interceptaran la trayectoria rectilínea predefinida; después de haberse filmado las diferentes situaciones y llevar a cabo un análisis de las observaciones se obtuvieron las siguientes conclusiones: 2a) la M.R. al encontrarse

con un obstáculo en su trayectoria, el cual puede ser tratado como una barrera, SIEMPRE lo sube y camina sobre su cara superior siguiendo la trayectoria rectilínea fijada en el alcance del refugio cuando no había obstáculos, hasta que llega a un borde; 2b) la M.R. al encontrarse en un borde de la cara superior de un obstáculo, teniendo entre éste y otro obstáculo una zanja, tiene 3 tipos de comportamiento: bajar por la zanja; saltar hacia otro obstáculo que esté dentro de su trayectoria rectilínea definida; tratar de encontrar otro camino en un rodeo de la cara superior del obstáculo.

Cabe hacer notar que un objeto puede ser visto como una barrera, y dos objetos conforman una zanja, de modo que al considerarse una zanja será porque la M.R. está sobre uno de ellos y tiene al otro objeto frente a ella como el otro extremo de la zanja; así, llamaremos profundidad de la zanja a la altura del objeto sobre el cual está la M.R. y altura del otro objeto o profundidad del otro extremo de la zanja a la altura del objeto que está frente a ella.

De acuerdo con los resultados obtenidos hasta este punto, se observa que existen acciones o tácticas que siempre se llevan a cabo (definición de la trayectoria; subir siempre una barrera, caminar sobre la cara superior de una barrera hasta un borde siguiendo la trayectoria rectilínea), y otras que su ejecución está definida por la métrica de la situación, específicamente en las alternativas de la forma de salvar una zanja (salto a otro obstáculo, bajar la zanja, rodear), siendo las condiciones métricas influyentes en la

toma de alternativas la altura del objeto y la distancia a la que está el otro objeto, o equivalentemente la profundidad y longitud de la zanja respectivamente.

Para las alternativas presentadas, en la forma en que una zanja era salvada, sólo se consideran en este trabajo la del salto a otro obstáculo, pasar a otro obstáculo y la de bajar zanja, dejándose la de rodeo por no haberse estudiado lo suficiente, pero sabiendo, de acuerdo con los datos hasta ahora obtenidos, que ésta se ejecuta como alternativa, cuando no puede realizar alguna de las otras 3 tácticas debido a las condiciones métricas.

La táctica de salto a otro obstáculo que está en la misma trayectoria rectilínea se estudió llevando a cabo una serie de experimentos que arrojaron como consecuencia de su análisis que es un salto balístico (el cual puede estudiado en base a la 2a. ley de Newton) y que existe una variedad de ejecución de acuerdo a la métrica de la zanja, y que por considerarse de gran interés se presenta su análisis y la obtención de un modelo matemático de esta táctica en el capítulo III.

La determinación de cuando el animal presenta la conducta de bajar una zanja(o hacer la respuesta de rodeo, sólo que en este trabajo no se presenta el estudio de esta conducta) es dependiente de la realización del salto balístico; esto es, si la M.R. no puede saltar al otro extremo de la zanja(obstáculo que tiene al frente), se decide por bajar hasta el fondo de la zanja. Por tanto, la

determinación de cuando el animal baja la zanja es dependiente de la aplicación del modelo de salto balístico presentado en el capítulo III.

Todos los experimentos mencionados, además de los tratados en el capítulo III (salto balístico), sirven para definir totalmente el modelo de esta conducta, por lo cual es posible llevar a cabo su desarrollo y simulación, lo cual es presentada en la sección siguiente.

II.3 IMPLEMENTACION Y SIMULACION DEL MODELO DEL ALCANCE DE UN REFUGIO POR LA M.R.

El modelo de alcance de un refugio de la M.R. está basado en la aplicación de Teoría de Esquemas que intenta explicar toda la compleja gama de acciones que involucra esta conducta de la M.R., y postular el procesamiento de información que debe realizar el Sistema Nervioso de estos animales en estas circunstancias. La Teoría de Esquemas propone que el Sistema Nervioso de los animales está organizado por estructuras de información que denominamos esquemas, los cuales permiten al animal definir la pauta motora adecuada para alcanzar un objetivo dado.

La simulación del modelo de la conducta de alcance de un refugio de la M.R. se llevó a cabo por medio de un programa en computadora digital, el cual se detalla a continuación.

El objetivo del programa es llevar a cabo la simulación de la conducta de alcance de un refugio de la M.R. teniéndose la capacidad de definir cualquier espacio tridimensional. Es decir, podrán ser definidos objetos en cualquier disposición y con dimensiones varias, teniéndose también la capacidad de definir la posición de la mantis en cualquier posición del espacio tridimensional, para, con este espacio definido, llevar a cabo la simulación de la conducta en cuestión. La evolución de la simulación se observará por medio de imágenes tridimensionales de color animadas en un monitor de resolución media(378x241 pixels) que es manipulado por un programa en computadora digital que se encarga de presentar en imágenes los resultados de la aplicación del modelo teórico; la definición del espacio a simular es interactiva y la define el modelador(usuário) a través de un cursor que aparece en un espacio tridimensional en una pantalla de graficación de color, y que es susceptible de moverse en las tres dimensiones y en ambos sentidos de cada una con la simple presión de teclas predefinidas de una terminal, para que en combinación con la aplicación de comandos(por medio de la presión de una tecla) sea posible definir objetos en cualquier posición y de cualquier tamaño(dentro de los límites de graficación), así como una mantis religiosa, cuya posición, al igual que los objetos, puede ser definida en cualquier parte del espacio tridimensional de graficación definido.

II.3.1. Consideraciones y restricciones del programa de simulación.

El espacio tridimensional de simulación en el programa cuenta con algunas restricciones, que son las siguientes:

- a) Los objetos a definir sólo son paralelepípedos.
- b) Todos los objetos deben estar sobre el "suelo"; ésto no permite definir objetos sobre otros.
- c) No es posible la existencia de intersección de objetos.
- d) Los objetos serán dibujados sólo en su contorno; esto es, como si fueran transparentes, y es con el fin de observar los objetos que se encuentran detrás de ellos.
- e) Debe existir por lo menos un objeto en el espacio.
- f) Es posible la existencia de una sola mantis religiosa.
- g) La posición inicial de la mantis religiosa debe ser definida sobre el "suelo"
- h) El movimiento de la mantis en la simulación sólo se lleva a cabo de derecha a izquierda.
- i) Los objetos cuya posición esté detrás de la inicial de la mantis religiosa son ignorados en la simulación.

EL resultado de estas restricciones se deben a las características con que se realizaron los experimentos empíricos(a, b, c, f,) y a una cierta facilidad para representar en forma gráfica tridimensional los resultados de la simulación del modelo(restricciones d,e,g,h,i).

II.3.2. Conformación del programa de simulación.

El programa de simulación fue desarrollado en el lenguaje de programación FORTRAN, así como macro-assembler de Z80. Fueron utilizadas dos bibliotecas, que son las siguientes:

-FORLIB: realiza funciones matemáticas.

-SDILIB: permite la manipulación de la pantalla de graficación mediante llamadas a rutinas desde FORTRAN; las funciones que permite SDILIB son las siguientes: escalar la pantalla de graficación; seleccionar uno de 16 colores para graficación; pintar un punto; pintar una línea; seleccionar una de las dos páginas de graficación para trabajo; desplegar en la pantalla una de las 2 páginas de graficación; borrar una de las dos páginas de graficación.

El hecho de que pueda llevarse a cabo la simulación por animación es debido a que la computadora cuenta con una interfase para la graficadora; esta interfase manipula dos tarjetas de memoria RAM de 48K bytes cada una, de tal forma que cada tarjeta de memoria es capaz de almacenar una imagen de la pantalla de graficación completa; de esta forma, cuando se dibuja, lo que se hace es modificar el contenido de la tarjeta de memoria, la cual a su vez es accesada por la interfase a cada instante, convirtiendo su contenido a

imágenes. La interfase nos permite seleccionar lo que llama página de trabajo y de graficación. La página de trabajo es la tarjeta de memoria sobre la cual nosotros estamos dibujando; la página de graficación es la que está accedando en cada momento y desplegando en la graficadora.

Si llamamos a cada una de las páginas 1 y 2, y consideramos que queremos graficar una secuencia de n cuadros que representan la discretización del movimiento de una escena en precisamente n tiempos, es posible simular el movimiento de la escena con el siguiente método:

```
-begin algoritmo
-selecciona página 1 como página de trabajo.
-selecciona página 2 como página de graficación.
-for i=1 to n do
  begin
    dibuja cuadro i.      %(esto se hace en página de trabajo)
    asigna: página de trabajo=página de display
              página de display=página de trabajo
              %(esto provoca borrar el cuadro i-1
              % y desplegar el cuadro i)
  end;
-end.
```

El método anterior es el usado para llevar a cabo la simulación de la conducta por medio de graficación animada.

La organización del programa de simulación de esta conducta se muestra en la figura F.II.1, y como se observa consta de 2 módulos principales: el de definición del espacio y el de simulación del espacio definido. La razón

por la que se particionó el programa fue debido a que, por haberse implementado en una microcomputadora, se tenía déficit de memoria RAM si se conceptuaba en uno solo.

En la figura F.II.2 y F.II.3 se muestran los diagramas de flujo de los módulos de definición y de simulación respectivamente. Como se observa el módulo de simulación es prácticamente el etograma de espacio de estados de la conducta de Objetos sin movimiento (sin incluir la respuesta de rodeo), mostrando con esto la utilidad de definir una conducta por medio de la teoría de esquemas, haciendo susceptible de simulación directa el comportamiento del robot o animal para el cual está definido el etograma.

El programa fue desarrollado en forma modular y con programación estructurada, mostándose de forma implícita en el programa fuente (ver APENDICE A) los diferentes módulos mostrados en las figuras F.II.2 y F.II.3, así como de las diferentes etapas que los comprenden. Todo esto, aunado a la documentación total del programa fuente, permite el estudio, seguimiento y comprensión de las técnicas utilizadas en la representación de un espacio tridimensional, representación de los objetos, representación de la mantis y movilidad de sus patas, que es lo que da la impresión de caminar.

Cabe hacer notar que todos los métodos son producto de programación heurística, por lo cual no hay referencia a artículo o técnica anteriormente publicado.

En el programa se contempla un sistema tridimensional como el de la figura FII.4, y la forma de obtener la representación de objetos tridimensionales, - así como de

otras técnicas utilizadas en el, son presentados a continuación.

Un punto 3-D está definido por tres coordenadas, pero para hacer posible su graficación en un plano, en este caso la pantalla de graficación, es necesario obtener una proyección de el para dar en el espacio bidimensional de graficación la apariencia de profundidad, completando de esta forma las 3 dimensiones a representar. Las ecuaciones siguientes nos dan las coordenadas cartesianas para representar un punto tridimensional:

$$XG = X + \text{COS}(\text{PSI} * Y) \quad \dots \text{(EII.1)}$$

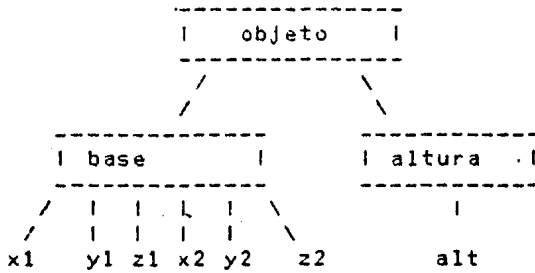
$$YG = Z + \text{SIN}(\text{PSI} * Y) \quad \dots \text{(EII.2)}$$

donde: X, Y, Z son las coordenadas del punto 3-D.

XG y YG son las coordenadas de graficación en el plano del punto 3-D.

PSI es el ángulo de proyección del punto tridimensional referido al sistema bidimensional de graficación.

Los objetos tridimensionales considerados en el programa son paralelepípedos, y observando las restricciones y consideraciones mencionadas en el punto II.3.1, quedan totalmente definidos dentro del espacio dimensional con 2 características: su base y su altura. Su base, a su vez, está definida por las coordenadas de de 2 esquinas contrarias. Por lo anterior, un objeto queda definido con la siguiente estructura:

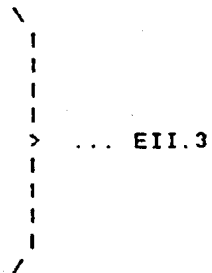


Con esta información es posible dibujar un paralelepípedo, lográndose esto por la unión de sus vértices por medio de líneas rectas; estas líneas rectas son tridimensionales y están definidas por 2 puntos también tridimensionales; al obtener la proyección de estos dos puntos con la aplicación de las ecuaciones EII.1 y EII.2 y dibujar una línea recta entre las proyecciones obtenidas se obtiene una línea recta dibujada en un espacio 3-D, formándolo por completo al repetir esto con cada una de las líneas de las caras del objeto.

Para representar la mantis religiosa en el espacio tridimensional se digitalizó un dibujo de vista de perfil del cuerpo completo, considerandose un eje cartesiano, cuyos ejes pasan en el cuello de la mantis(figura FII.5). La posición de la mantis en el espacio tridimensional es considerando en el punto (XM, YM, ZM), el cual está situado en el origen del eje cartesiano a los cuales están referidos las coordenadas de su cuerpo. Este dibujo de la mantis es considerado en el espacio tridimensional como si estuviera contenido en un plano paralelo al X-Z, es decir, es 'plano',

no tiene grosor o profundidad; así, cada uno de los puntos del dibujo del cuerpo tendrán las siguientes coordenadas en el espacio tridimensional:

- (x1+XM, YM, z1+ZM),
- (x2+XM, YM, z2+ZM),
- (x3+XM, YM, z3+ZM),
- ⋮
- (xn+XM, YM, zn+ZM)



de los cuales es obtenida su proyección y son dibujados.

Para obtener una gráfica de la mantis en diferentes posiciones se tiene asociado al dibujo digitalizado un ángulo(definido este ángulo en el plano X-Z de la figura F.II.4) y un valor lógico; con estos valores del ángulo y del valor lógico se aplican transformaciones sobre los datos originales, para obtener nuevas (x1,z1), (x2,z2), (xn,zn) que sustituidas en la expresión, EII.3 nos dan, como ya se dijo, diversas posiciones de la mantis en el espacio tridimensional. El ángulo indica el valor angular, considerando el plano cartesiano al cual están referidas las coordenadas del dibujo de la mantis, al que se debe rotar cada uno de los puntos que conforman su cuerpo. El valor lógico indica si se obtiene la imagen espejo con respecto a uno de los ejes del plano cartesiano considerado para la digitización del dibujo. Con la combinación de ambas transformaciones y con las coordenadas tridimensionales de

la mantis es posible colocarla en posiciones como caminar en el suelo, subir o bajar por una cara lateral del objeto, caminar sobre la cara superior del objeto o en el aire simulando un salto.

Para llevar a cabo el movimiento de la mantis se consideraron 2 posiciones de 3 de las 4 patas torácicas de la mantis, esto para poder dibujar la forma en que camina la mantis, es decir:

-movimiento hacia adelante lateral hacia afuera de la pata delantera derecha.

-movimiento hacia adelante lateral hacia afuera de la pata trasera izquierda.

-movimiento hacia adelante lateral hacia afuera de la pata delantera izquierda.

-avance logrado por el impulso hacia adelante de la pata trasera derecha.

Considerando estos movimientos, y en combinación con las transformaciones que actúan en el cuerpo de la mantis ya descritas, es posible dar el efecto de animación de las acciones de subir y bajar objetos, caminar sobre el suelo y caminar sobre otros objetos.

Con lo anteriormente explicado y con la manipulación de las propiedades de los objetos (alto, ancho, grosor y posición) es posible llevar a cabo la simulación de esta conducta en la forma de imágenes tridimensionales animadas.

II.3.3. Forma de uso del programa de simulación.

La simulación del modelo de la conducta de chantlitaxia se lleva a cabo por dos programas, el primero llamado DEFINEM.COM se encarga de la definición del espacio a simular en una forma interactiva, creando al final de la definición un archivo de datos llamado DEFINE.MAN que describe totalmente el archivo definido para su simulación. El segundo programa llamado SIMULAM.COM lleva a cabo la simulación del espacio definido, accedando para esto el archivo de datos DEFINE.MAN generado por DEFINEM.COM.

La razón por la que conceptuó en dos programas la simulación del modelo se debe a las limitantes en memoria RAM que tiene la computadora CROMEMCO(64Kb) y a su vez el alto consumo de memoria por parte de las bibliotecas de graficación de la interface SDI(16 Kb aproximadamente), teniéndose con esto solamente 36 Kb para área de usuario, ya que el sistema operativo CDOS ocupa entre 12 y 15 Kb.

Estos dos programas son llamados desde un archivo de comandos llamado SIMCHAN.CMD para darle al usuario la facilidad llevar a cabo la definición y simulación de una situación ejecución de un solo comando.

La forma de operar el programa es la siguiente.

Teclear el nombre del programa, es decir:

SIMCHAN<return>

A continuación aparecerá el espacio tridimensional en la pantalla (figura FII.6), observándose un cuadrado que se enciende y apaga, al cual llamaremos cursor.

A partir de este momento se procederá a hacer la definición del espacio a simular, lográndose esto con el movimiento del cursor y el uso de comandos. En la parte superior de la pantalla de graficación se tienen indicaciones que guían y ayudan al usuario en la definición del espacio. Se observa que en la parte superior izquierda aparece la siguiente leyenda:

CURSOR: <, >, <-, ->

lo cual indica que el cursor puede moverse con la presión de las cuatro teclas indicadas; el movimiento que ocasiona cada una de las teclas es como sigue:

- > mueve el cursor hacia la derecha(X positivas).
- <- mueve el cursor hacia la izquierda(X negativas).
- > mueve el cursor hacia "adentro"(Y positivas).
- < mueve el cursor hacia "afuera"(Y negativas).

En la parte superior derecha aparece la leyenda:

COMANDO: Objeto, Mantis

lo que indica que para definir un objeto o una mantis basta teclear la primer letra del comando('O' , 'M').

Antes de pasar a explicar cada uno de los comandos es conveniente hacer ciertas aclaraciones.

-Para definir un objeto, se definirá primero su base y luego la altura. La base será definida a su vez por dos esquinas contrarias dadas en cualquier orden.

-Al definir la mantis religiosa, se dará por finalizado la etapa de definición, pasándose a simular el espacio definido.

-El programa mandará mensajes de error en la pantalla de graficación y/o hará sonar una campana de la terminal cuando:

- a) se teclee un comando inválido.
- b) se defina un objeto dentro de otro.
- c) se desee mover el cursor a una posición fuera del espacio.
- d) se deje de definir algún objeto y se quiera presentar a la mantis religiosa.
- e) se defina la mantis religiosa dentro de un objeto.
- f) sea mal definida una base(que las 2 esquinas forman una línea).
- g) se defina la altura de un objeto con un valor de cero.

Con lo anterior el programa no permite hacer definiciones erróneas, apegándose a las restricciones consideradas en el punto II.3.1.

DEFINICION DE UN OBJETO.

La definición de un objeto se lleva a cabo siguiendo la secuencia que a continuación se muestra.

-01)teclear O (definición de Objeto): se observa que en la parte superior de la pantalla aparece'

CURSOR:<, >, <-, -> OBJETO:Listo para definir esquina 1

lo cual indica que se tienen las teclas mostradas a continuación de 'CURSOR', para provocar su movimiento. La leyenda de la parte derecha indica que se está definiendo un objeto, y que cuando el cursor esté en el lugar donde se desee definir la esquina no.1 de la base se teclee L.

-02)mover el cursor con las teclas que aparecen a continuación de 'CURSOR:' al lugar donde se desee quede definida la primer esquina de la base.

-03)teclear L para definir la primer esquina: se observa que después de teclear L se dibuja un punto de color en la posición definida como esquina 1 y aparece lo siguiente en la parte superior de la pantalla:

CURSOR:<, >, <-, -> OBJETO:Listo para definir esquina 2

el cual tiene el mismo significado que en paso (01), pero ahora para definir la esquina 2 de la base, la cual

debe de ser contraria a la 1, ya definida.

-04)mover el cursor a la posición donde se desee definir la esquina 2 de la base.

-05)teclear L para definir la segunda esquina; se observa que después de teclear L se dibuja la base del objeto definida y aparece en la parte superior de la pantalla de graficación lo siguiente:

```
CURSOR: ^ I          OBJETO:Listo para definir altura
         | V
```

significando que ahora el cursor puede moverse hacia "arriba" o hacia "abajo" con presionar las teclas mostradas a continuación de "CURSOR:", ya que ahora se va a definir la altura del objeto.

-06)mover el cursor a la altura deseada del objeto, utilizando las teclas que están a continuación de "CURSOR:"

-07)teclear L para definir la altura del objeto tomando la posición que tenga el cursor; se observa que después de teclear L se dibuja el paralelepípedo definido y en la parte superior de la pantalla aparece la leyenda:

```
CURSOR: <, >, <-, ->  COMANDO:Objeto,Mantis
```

Indicándose con esto que se ha finalizado la definición de un objeto y se está nuevamente en posibilidad de definir un nuevo objeto o la posición de la mantis religiosa.

Con la forma anteriormente descrita es posible definir cualquier numero de objetos, creando de esta manera las barreras y zanjas a considerar en la simulación.

En las figuras F.II.7 a F.II.10 se muestra la evolución de la definición de un objet

DEFINICION DE LA MANTIS RELIGIOSA.

Para definir una mantis religiosa y dar por terminada la definición del espacio se muestra a continuación la secuencia a seguir.

-M1)teclear M (indica definición de Mantis); se observa que en la parte superior de la pantalla aparece:

CURSOR:<, >, <-, -> Mantis:L para definir posición

lo cual indica que el cursor puede ser movido con las teclas que aparecen a continuación de "CURSOR:" y que se está en el comando definición de mantis.

-M2)mover el cursor a la posición la que se desee definir a la mantis.

-M3)teclear L para que la mantis sea definida donde está el cursor, observándose que la mantis es dibujada en en tal posición y aparece en la parte superior de la pantalla:

PRINCIPIA SIMULACION.

lo cual indica que se ha terminado la etapa de definición del espacio y se procederá a simular el modelo

conceptual de la respuesta de la M.R. ante la situación que se acaba de describir.

Una vez ocurrido esto, la primer escena de la simulación mostrará una línea recta que va de la posición inicial de la mantis al objeto que se considera el mejor refugio en su campo visual; dicha línea representa la trayectoria a seguir. En los cuadros posteriores se observará la evolución del comportamiento de la mantis religiosa ante el espacio definido.

Las figuras F.II.11 y F.II.12 muestran un espacio definido y la evolución de la simulación de éste por medio de una fotografía estroboscópica, respectivamente.

Cuatro simulaciones más que presentan todas las acciones motoras posibles a realizar por el modelo son mostradas en las figuras F.II.13 a F.II.20 por medio de fotografías estroboscópicas.

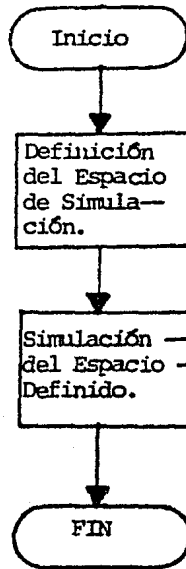


Figura F.II.1. Módulos del Programa de Simulación de la conducta Búsqueda de Refugio (Chantlitaxia) de la Mantis Religiosa.

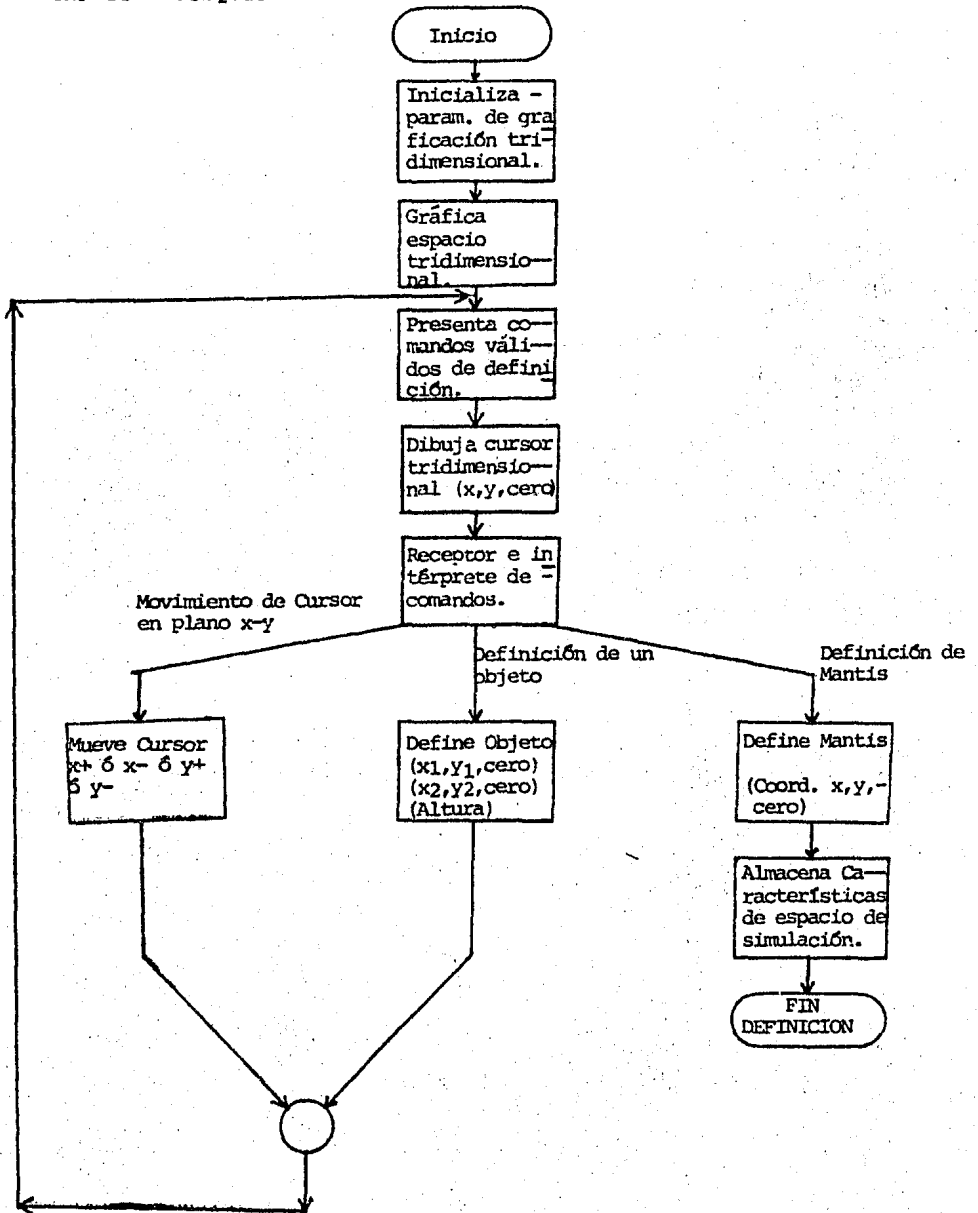


Figura F.II.2

Diagrama de bloques del Módulo de Definición del espacio de Simulación para el modelo de Chantlitaxia.

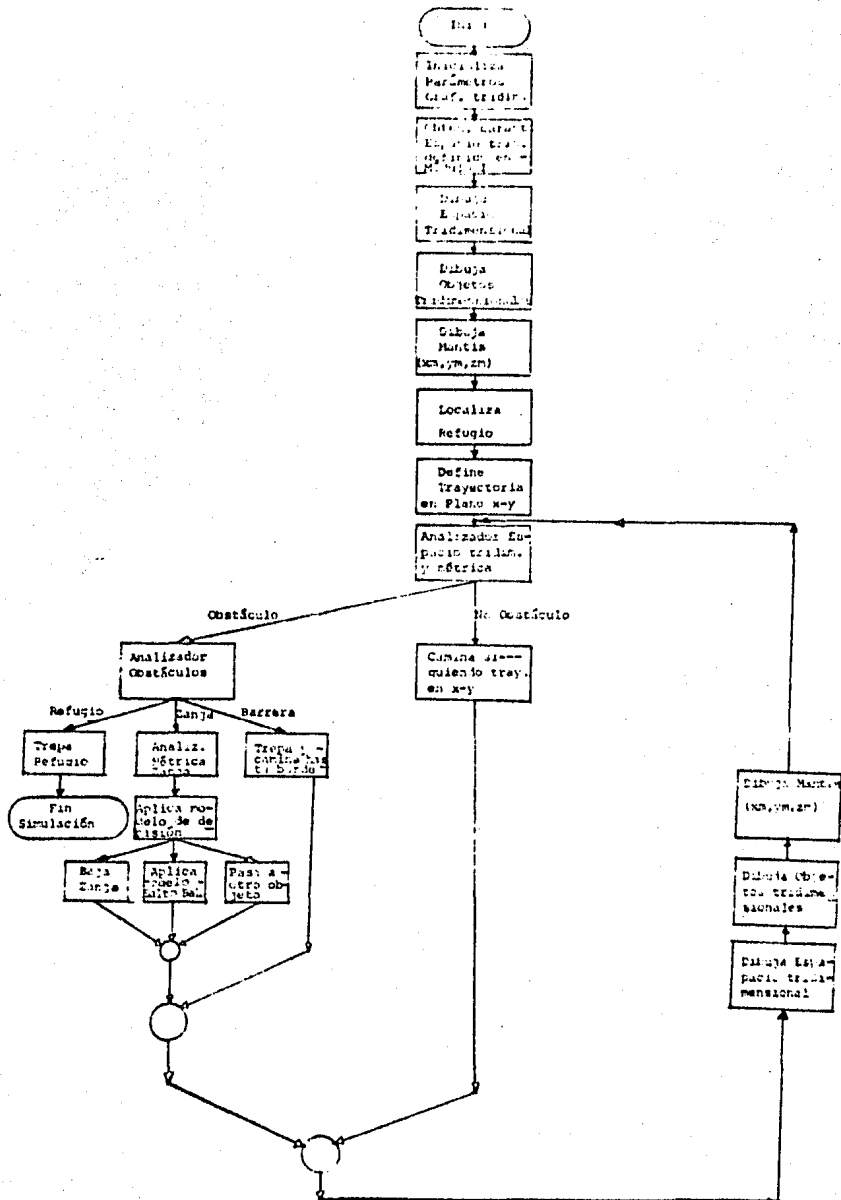


Figura F.21.3. Diagrama de Bloques del Módulo Simulación del espacio definido para el modelo Búsqueda de Refugio (Chatilina).

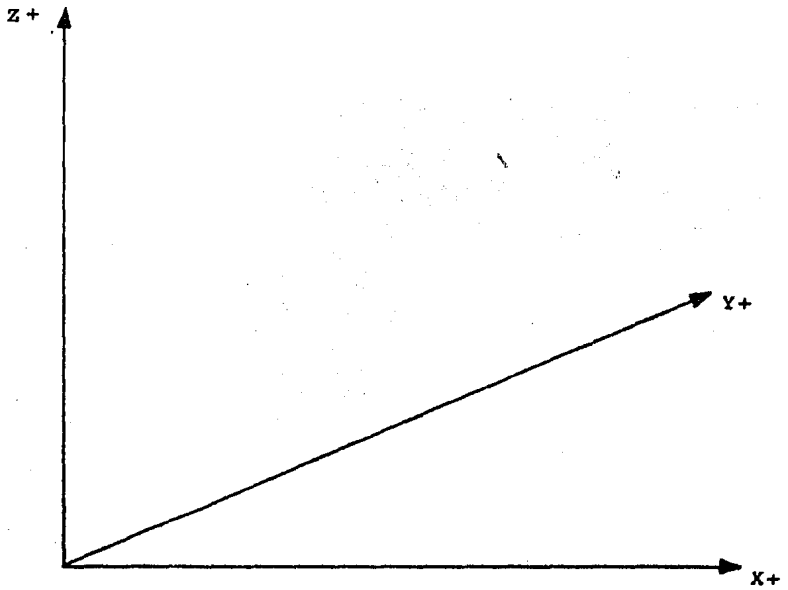


Figura F.II.4

Sistema de Ejes considerados en el programa de simulación de la Conducta Busqueda de -refugio de la mantis Religiosa.

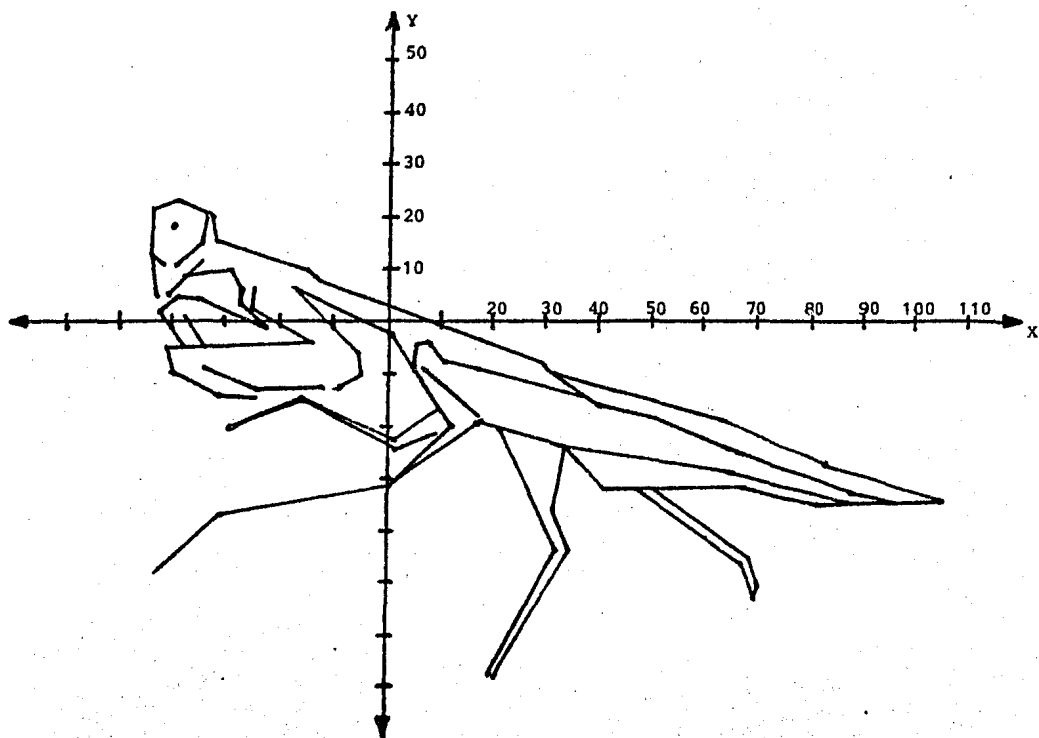


Figura F.II.5 Mantis Religiosa. Dibujo digitalizado.

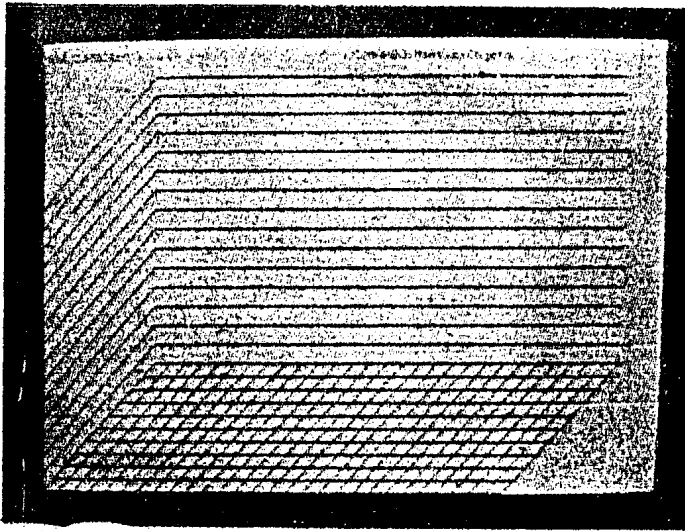


FIGURA F.II.6

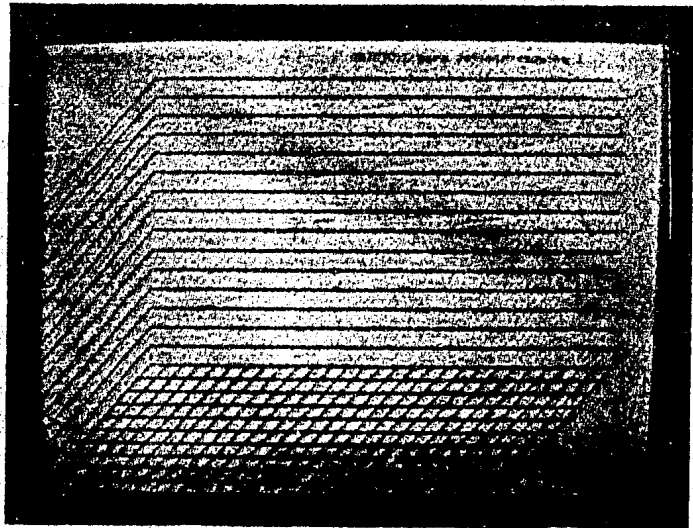


FIGURA F.II.7

Definición de Objeto. (F.II.6) Espacio de definición.
(F.II.7) Definición de esquina 1; nótese el mensaje que se tiene en la leyenda superior derecha, así como la posición del cursor en la última celda del eje Y y primera del X.

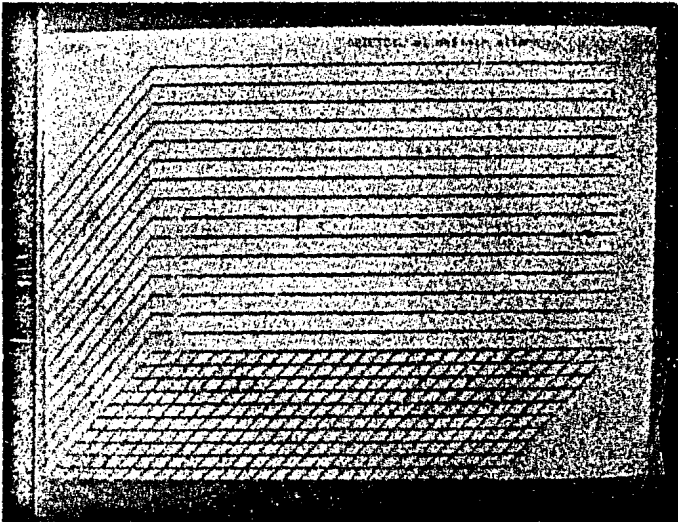


FIGURA F.II.8

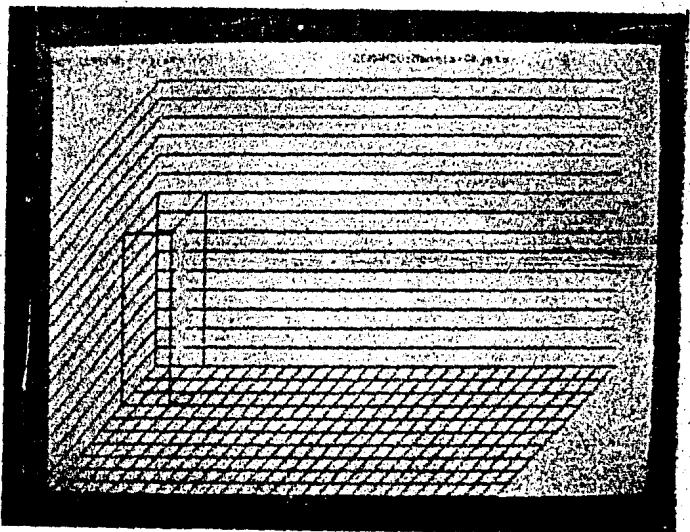


FIGURA F.II.9

Definición de Objeto (Continuación). (F.II.8) Definición de altura dada la base del Objeto; nótese el cursor en $X=2$, $y=9$, $Z=10$ y la leyenda de "L al definir altura" así como los movimientos del cursor de arriba - abajo en la parte superior de la pantalla. (F.II.9) Objeto definido.

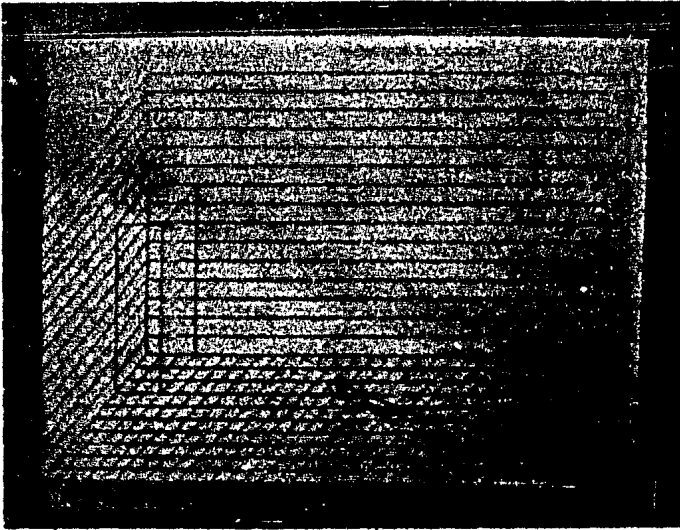


FIGURA.II.10

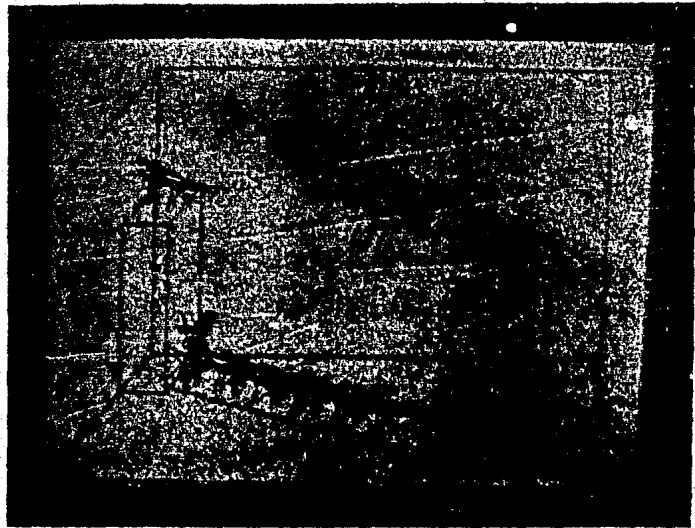


FIGURA.E.II.11

Ejemplo de simulación 1. (F.II.10) Espacio definido para su simulación; solo existe un objeto que es el refugio. (E.II.11)- Simulación del espacio; se observa una línea recta que une la posición inicial de la mantis con el refugio, así como la serie de acciones necesarias para alcanzar el refugio (Caminar hasta el refugio y trepar).

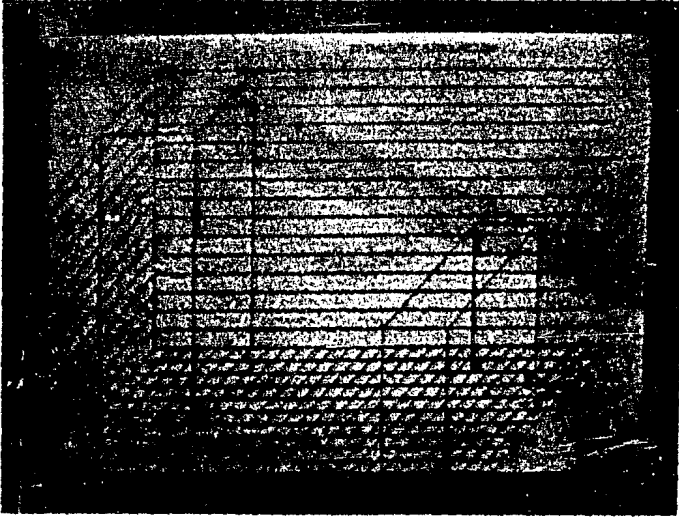


FIGURA F.II.12

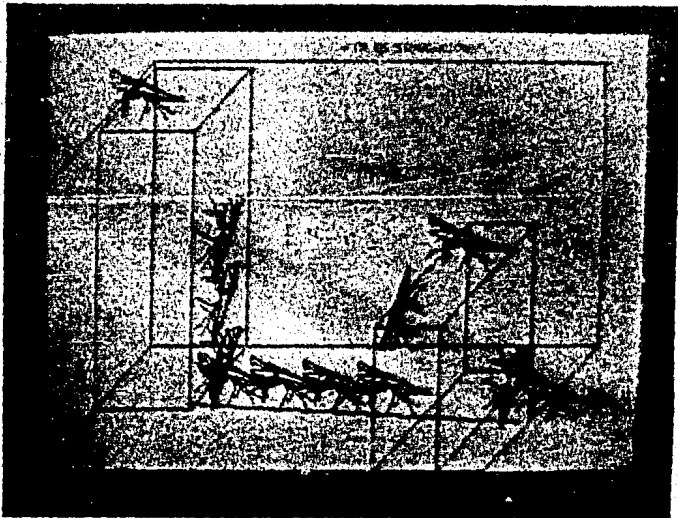


FIGURA F.II.13

Ejemplo de simulación 2. (F.II.12) Espacio definido con 2 objetos, uno de los cuales es el refugio . (F.II.13) Simulación; se observa la trayectoria a seguir hasta el refugio; se aproxima hasta el primer obstáculo y lo trepa, camina sobre la cara superior hasta el borde, aplica el modelo de salto balístico y no puede llevarlo a cabo, por lo que baja el obstáculo, camina hacia el refugio y lo trepa.

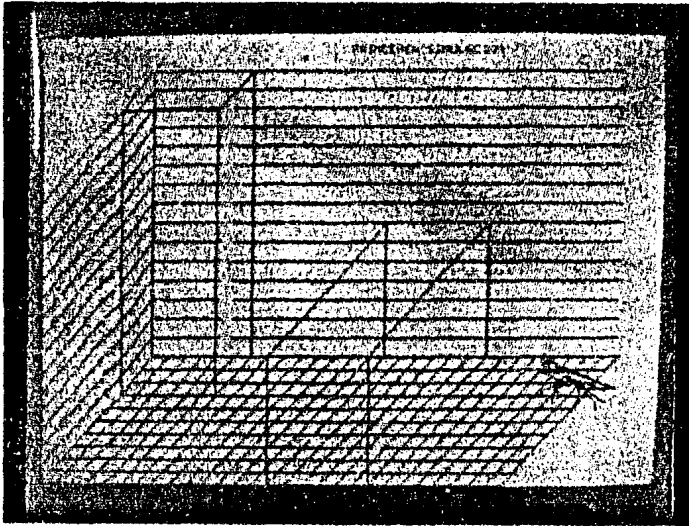
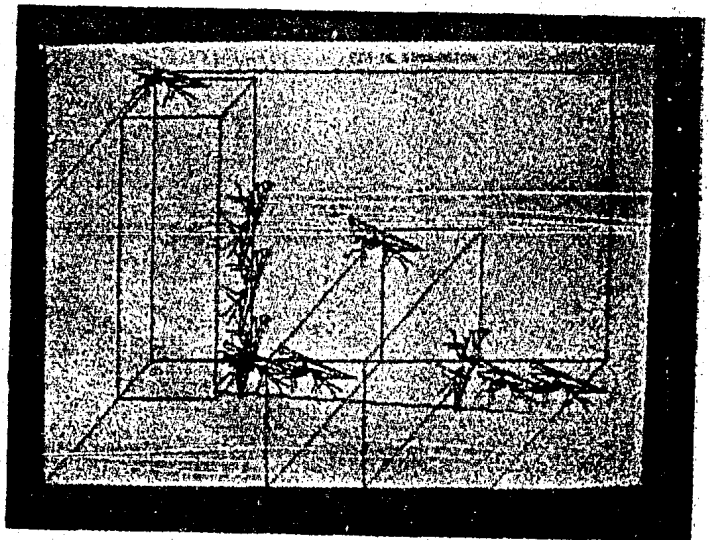


FIGURA F.II.14

FIGURA F.II.15



Ejemplo de Simulación 3. (F.II.14) Espacio con dos Objetos.-
 (F.II.15) Simulación; se aproxima al primer obstáculo, lo sube y camina sobre su cara superior; al estar ante una zanja aplica modelo de salto balístico y no lo realiza, dejándose caer al suelo por una altura pequeña del objeto, se dirige hacia el refugio y lo trepa.

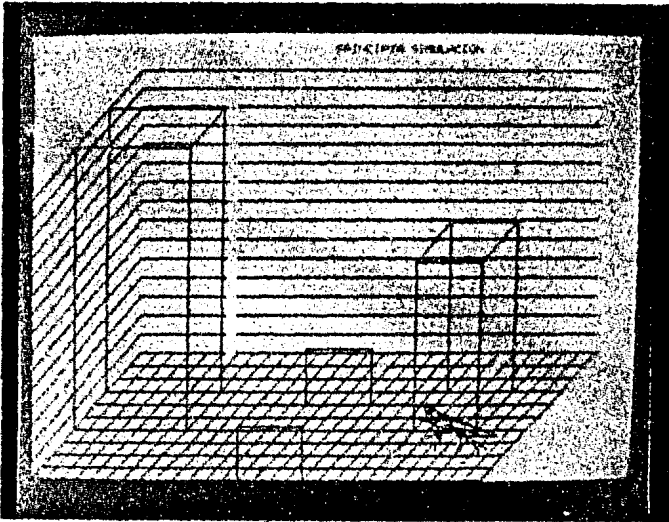


FIGURA F.II.16

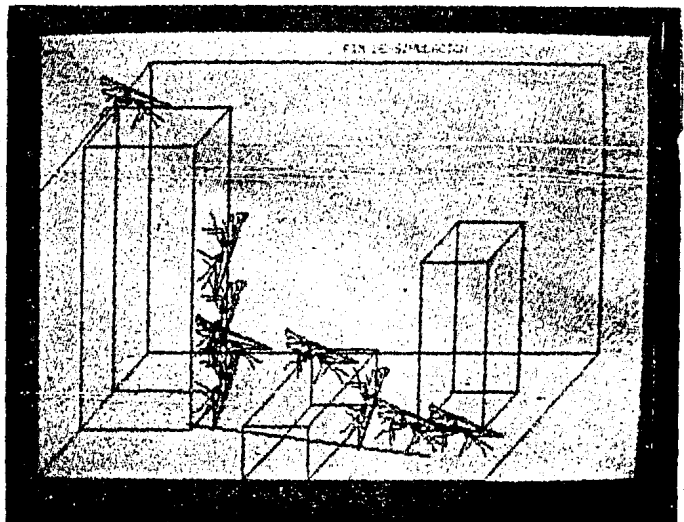
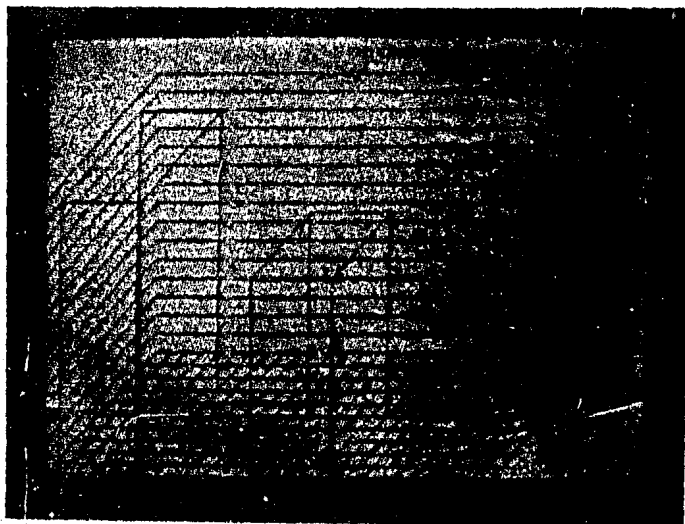
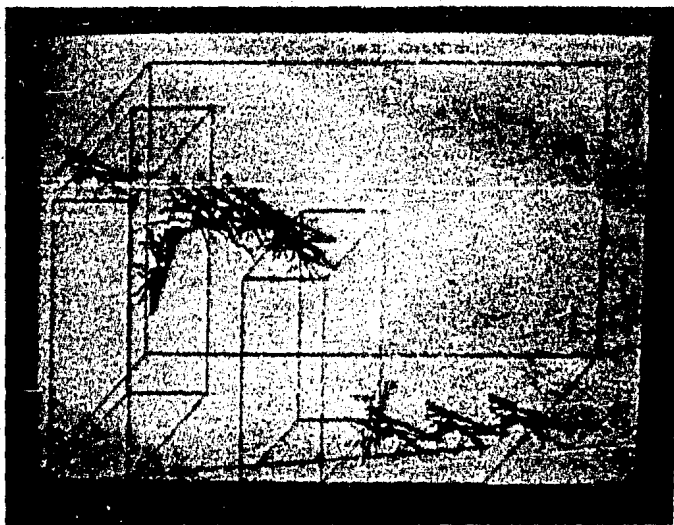


FIGURA F.II.17

Ejemplo de Simulación 4. (F.II.16) Espacio definido con 3 objetos. (F.II.17) Simulación; el objeto de más a la derecha está detrás de ella, así que lo ignora; se dirige al obstáculo, lo sube y camina sobre su cara superior; aplica modelo de salto balístico no lo realiza y debido a la pequeña distancia entre los dos objetos sólo extiende las patas para salvar la zanja y trepa el objeto que es identificado como refugio.



F.II.18



F.II.19

Ejemplo de Simulación 5. (F.II.18) Espacio definido con 2 objetos. (F.II.19) Simulación; se aproxima al primer objeto, lo trepa y camina por su cara superior; se encuentra ante una zanja, aplica modelo de salto balístico, lo realiza y trepa por el objeto identificado como refugio.

CAPITULO III .

SALTO BALISTICO .

OBTENCION DE MODELO MATEMATICO .

III.1. INTRODUCCION.

En el estudio de la conducta de la búsqueda y alcance de un refugio por la mantis religiosa se observó que cuando la mantis se encuentra ante una zanja, esta presenta una de 4 diferentes conductas para salvarla: 1) bajar la zanja; 2) pasar al otro extremo de la zanja; 3) salto; 4) rodear. Como ya se dijo en el capítulo II, la elección de las alguna de las conductas para salvar la zanja depende de las medidas de ésta, pero la que se realizaba con mayor frecuencia y con mayor número de situaciones métricas de la zanja es la de salto. Por esta razón se decidió llevar a cabo estudios para observar bajo que circunstancias métricas se lleva a cabo el salto y sus características.

El salto de la Mantis Religiosa lo hemos denominado salto balístico debido a que la trayectoria seguida por el cuerpo del animal asemeja una bala al ser disparada. Esta conducta es tratada separadamente de las acciones alternativas involucradas en el alcance de un refugio por presentar por sí misma un gran interés por las diversas tácticas, variaciones en el impulso y el ángulo de despegue, que puede usar el animal ante diversas dimensiones de anchura y profundidad de la zanja para alcanzar el objetivo o refugio.

Los estudios etológicos del salto de la Mantis Religiosa, al igual que los descritos en el capítulo II, se complementaron con estudios teóricos sobre coordinación visuomotora que se han desarrollado con respecto a la rana[11]

A continuación se presentan los estudios etológicos realizados, así como el modelo matemático obtenido del análisis de ellos.

III.2. Modelo Matemático del Salto Balístico.

Para definir las diversas tácticas que emplea la mantis religiosa en el salto balístico ante diferentes circunstancias métricas de las ranjas, se le estudió variando la altura y la distancia entre el borde donde se encuentra el animal y el objetivo(figura FIII.1). La técnica utilizada en la cuantificación de los datos observados se fundamenta en la segunda ley de Newton para la caída libre de los cuerpos. Si un objeto se lanza con una velocidad v y un ángulo θ , podemos descomponer a dicho movimiento en componentes en el eje X y en el eje Y en la siguiente forma:

$$v_x = v \cdot \cos(\theta) \quad \dots \quad \text{EIII.1}$$

$$v_y = v \cdot \sin(\theta) - g \cdot t \quad \dots \quad \text{EIII.2}$$

cuya integración nos da las siguientes ecuaciones:

$$x = v \cdot \cos(\theta) \cdot t \quad \dots \quad \text{EIII.3}$$

$$y = v \cdot \sin(\theta) \cdot t - g \cdot t^2 / 2 \quad \dots \quad \text{EIII.4}$$

con lo cual se tiene definido el movimiento en los ejes X e Y en función del tiempo, y ángulo y velocidad iniciales de lanzamiento. Si despejamos a t de la ecuación EIII.3 y sustituimos en la ecuación EIII.4 obtenemos:

$$y = a_1 * x + a_2 * x * x \quad \dots \text{EIII.5}$$

donde: $a_1 = \tan(\theta)$

$$a_2 = - \frac{g}{2 * (v * \cos(\theta))^2}$$

o también:

$$y = a_0 + a_1 * x + a_2 * x * x \quad \dots \text{EIII.5.1}$$

donde: $a_0 = 0$

con lo cual tenemos ahora definido el movimiento para las ordenadas solamente en función de la velocidad y ángulo iniciales así como del valor del objeto en el eje horizontal X. esto es, eliminamos la variable tiempo de la ecuación.

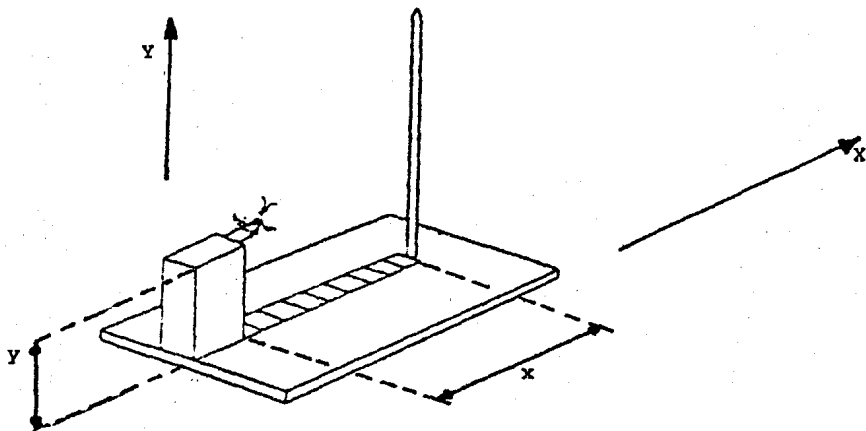


Figura F.III.1. Arena utilizada en la realización de experimentos para la obtención del modelo del salto balístico

Al saltar el animal de la plataforma hacia el objeto que tiene al frente se procesó la información para obtener

una serie de puntos x,y de la trayectoria respectiva e implementando un programa de computadora (el cual se muestra en el Apéndice B) que lleva a cabo en forma completa un análisis de regresión no-lineal utilizando el método de mínimos cuadrados, se ajustaron los datos de la trayectoria, producto del salto, a una parábola que tiene la forma de la ecuación EIII.5.1, obteniendo de esta forma valores de θ y v (ángulo e velocidad de impulso respectivamente) con que el animal se lanzó hacia el objetivo. El modelo desarrollado permite, además, obtener en forma gráfica en una pantalla de color la trayectoria del salto y la curva parabólica ajustada a ésta; cabe mencionar que los grados de ajuste a la trayectoria están en el rango $(0.89, 0.99)$ haciéndose notar que los datos ajustados cumplen en un alto grado con una parábola.

La figura F.III.2 muestra los resultados de experimentos empíricos del salto de la M.R. a diferentes distancias entre animal y objeto, repitiéndose estos experimentos para animales en diferentes estadios de desarrollo ontogenético (la M.R. tiene a lo largo de su ciclo de vida de 6 a 8 estadios dependiendo de la especie, siendo un estadio el tiempo que hay entre muda y muda, caracterizándose que a medida que ocurre una muda el animal presenta un crecimiento corporal y una maduración de órganos sexuales), observándose los cambios que ocurren en la velocidad y el ángulo del salto balístico cuando se coloca el objetivo a diferentes distancias de la mantis.

III.3. Resultados.

En la figura F.III.2 se puede observar que para saltos a diferentes distancias, estos animales no cambian la velocidad del salto en función de la distancia del objetivo, sino que sólo disminuyen el ángulo del salto dependiendo de la distancia del objetivo. Se aprecia también en esta gráfica que en cada uno de los estadios, esta disminución de salto se aproxima a una línea recta con pendiente negativa, teniéndose por lo tanto, una pendiente diferente para cada estadio. Ahora, es claro que la pendiente en cada uno de los estadios depende de la fuerza muscular que tiene el animal en ellos, por tanto un animal que pertenezca al estadio uno(entre el nacimiento y muda 1)tiene menor fuerza que uno que pertenezca al estadio dos(entre muda 1 y muda 2) y mucho menor a otro que pertenezca al estadio 3(entre muda 2 y muda 3), y la propiedad que nos dice a que estadio pertenece un animal es su peso. La dependencia de cómo se modifica el ángulo del salto en función de la distancia del animal al objeto y del peso del animal, como se observa en la figura F.III.4, se puede definir por medio de la siguiente fórmula, que fue obtenida por medio del ajuste de los datos empíricos a una línea recta:

$$O(w, d) = m \cdot d + 79.8$$

... EIII.6

$$\text{donde } m = 5.13(1 - e^{-6.62 \cdot w})$$

donde w es el peso del animal y d es la distancia de la mantis al objetivo. Nótese que la pendiente de relación varia dependiendo del peso del animal, lo cual muestra que mientras más grande y pesado sea el animal, menor será la disminución del ángulo del salto ante un cambio de distancia dado. Al disminuir el ángulo del salto, el animal intenta incrementar la distancia del salto en el eje X. La variación de esta distancia con el ángulo cuando el animal se encuentra a una altura dada se presenta en la figura F.III.5 y se puede representar por medio de la siguiente fórmula que es obtenida por el ajuste de los datos a una línea recta con la distancia dx de la plataforma al objetivo y el ángulo θ obtenido del ajuste de la trayectoria de este salto:

$$dx = -0.1566 \cdot \theta + 17.796$$

... EIII.7

lo cual indica que mientras menor sea el ángulo del salto, mayor es la distancia que alcanza el animal en el eje

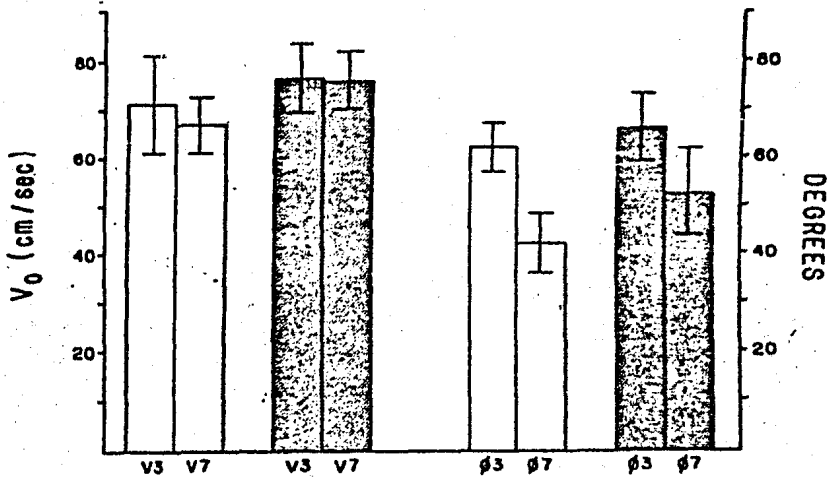


Figura F.III.2. Gráfica que muestra los cambios que ocurren en la velocidad y el ángulo de los experimentos analizados del salto balístico.

El salto balístico además de depender de la distancia al objetivo, depende también de la altura de la plataforma con respecto al objetivo; mientras mayor sea la altura de la plataforma, la distancia a la cual puede saltar el animal también aumenta, hasta un límite lo cual es mostrado en la figura F.III.3; en esta figura se observa la relación que existe entre las variaciones de altura de la plataforma con la distancia del salto que realizó en los experimentos, notándose una clara forma exponencial de esta gráfica. Esta relación puede expresarse por medio de la siguiente fórmula:

$$dsal = dmax - dmax * e^{-k * h} \quad \dots \text{EIII.8}$$

donde dsal es la distancia del salto; dmax es la

distancia máxima a la que el animal salta, la cual depende de su tamaño, esto es, entre más grande el animal mayor la distancia máxima a la que puede llegar; h es la altura de la plataforma; y k es la constante de crecimiento exponencial. El valor de k para los estadios 1 y 4 son 0.25 y 0.16 respectivamente, según observaciones y mediciones hechas en el desarrollo ontogenético del animal. Asimismo, el límite de altura en el objetivo al cual salta el animal se incrementa en función de la altura de la plataforma y la distancia de la plataforma al objetivo. Estos límites varían en función del tamaño del animal, siendo que entre mayor sea el animal, aumentan tanto su límite inferior como superior de altura. Esta relación se puede expresar por la siguiente fórmula:

$$h_{min}=0.6486*h - 3.5429 \quad \dots \text{EIII.9}$$

para el estadio 1; y

$$h_{min}=0.3600*h - 0.55 \quad \dots \text{EIII.10}$$

para el estadio 2;

donde h_{min} es la altura mínima en el objetivo a la que el animal salta; y h es la altura de la plataforma.

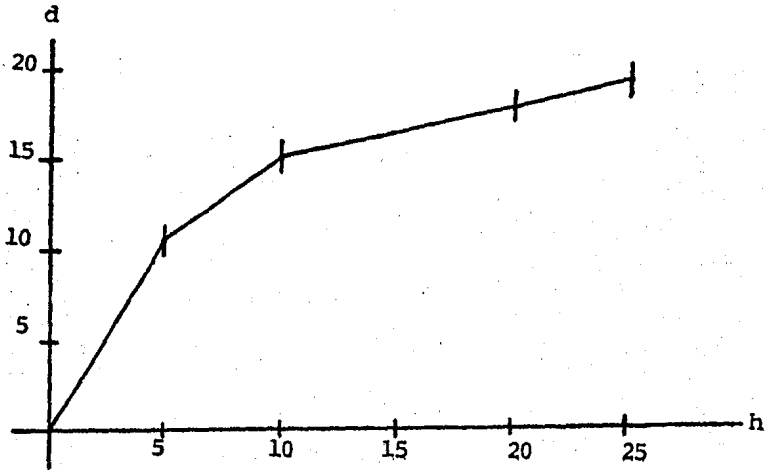


Figura F.III.3. Gráfica que muestra la relación existente entre la altura del objetivo y el salto balístico.

En conclusión, cuando el animal se encuentra que tiene que saltar una zanja para llegar a su objetivo, la decisión del salto depende de la relación de la altura del animal con respecto a la altura del objetivo; si el animal puede saltar, lo hace con velocidad constante y el ángulo depende de la distancia del objetivo.

Esto puede verse en el diagrama de bloques que se muestra en la figura F.III.6 observándose el flujo de operaciones llevar a cabo la evaluación de condiciones existentes cuando la M.R. está ante una zanja y tomar la decisión adecuada para librar este obstáculo

El modelo teórico de salto balístico, descrito en este capítulo, es el que se utiliza en la simulación de búsqueda de refugio presentada en el capítulo II.

C A P I T U L O I V

S I M U L A C I O N D E L A R E S P U E S T A
D E A T A Q U E H A C I A U N A P R E S A
D E L A M A N T I S R E L I G I O S A

IV.1 INTRODUCCION

La mantis religiosa es un insecto que permanece inmóvil en espera de una presa que quede a su alcance, a veces perfectamente oculta gracias a su capacidad de mimetismo. Una vez que ha identificado a una presa la mantis la sigue con movimientos de la cabeza y sin hacer movimiento alguno del tronco; en el momento en que la presa está a una distancia específica, la M.R. la atrapa por medio de sus patas delanteras, las cuales están cubiertas de espinas entre el fémur y la tibia que permiten inmovilizar a la presa para poder ingerirla lentamente.

La mantis religiosa es un animal carnívoro ya que se alimenta principalmente de insectos pequeños como avispas, chapulines, abejas, moscas, grillos, mariposas, cucarachas y, dependiendo de su estado de saciedad, ataca lagartijas y aún mamíferos como ratones pequeños.

La mantis religiosa captura su presa por una rápida acción de extensión y aprensión de las patas delanteras. La presa puede ser capturada en una variedad de ángulos con respecto al eje del cuerpo de la mantis, obteniendo

el éxito del 85 al 90 por ciento de sus intentos. La explosión muscular que impulsa las patas delanteras es tan rápida que el mismo animal es incapaz de detener este movimiento durante su ejecución. Mediciones de laboratorio muestran que este movimiento se lleva a cabo en el rango de 50 a 70 milisegundos como máximo. Esto indica que la mantis debe hacer una estimación previa de la dirección y distancia de la presa relativa a su propio eje del cuerpo, activando el patrón de excitación dirigido a los varios grupos de músculos conectados con las patas delanteras para que sobre el comando de "atrapa" estos músculos se contraigan dirigiendo el movimiento hacia la presa.

IV.2. MODELO MATEMATICO.

Se han examinado los sistemas que la M.R. activa para atrapar un insecto bajo varias condiciones [05] Mittelstaedt llevo a cabo una serie de experimentos que lo llevaron a postular un modelo de esta conducta. Este autor uso el aparato mostrado en la figura F.IV.1. que permite controlar los parámetros que intervienen en esta respuesta. La cabeza de la M.R. puede rotar libremente y su posición relativa con respecto al eje de su cuerpo (fijado) es registrada por un apuntador en la escala horizontal. Una mosca viva en una aguja es presentada a la M.R. con varios ángulos, y los ángulos

horizontales tanto de la mosca como de la cabeza de la mantis relativos al eje del cuerpo de la misma son medidos en la escala. Una mantis hambrienta sigue a la mosca muy de cerca haciendo movimientos de la cabeza que dependen de los movimientos de la mosca.

Si la presa esta cerca de la distancia de captura, la cabeza de la mantis sigue sus movimientos suave y continuamente; si la presa esta muy alejado del rango de las patas frontales, la cabeza de la mantis la sigue en series de movimientos bruscos. Mediciones en condiciones de equilibrio (cuando la mosca y la cabeza no están en movimiento) en un número de posiciones, muestran que el ángulo Y entre los ejes de la cabeza y el cuerpo (Figura F. IV. 2) es menor que el ángulo Z entre la dirección de la presa y el eje del cuerpo por una cantidad proporcional a la desviación de la presa al eje del cuerpo, al menos para los movimientos suaves que preceden a un ataque. Mientras que, cuando la presa esta adelante de la línea recta del cuerpo de la mantis, ambos ángulos son cero y la línea de fijación con respecto a la mantis es cero cayendo directamente en la de la presa; cuando la presa es movida a ángulos mayores, la línea referida a la mantis se retrasa de la de la presa por una cantidad proporcionalmente mayor. A esta desviación de la cabeza de la línea real de la presa se le llama DEFICIT DE FIJACIÓN(X). Análisis rigurosos muestran que X tiene un

valar constante y es aproximadamente el 15 por ciento de Z para los movimientos que preceden inmediatamente al ataque. El sistema usado por la mantis en el direccionamiento de su ataque pudo ser definido una vez que se encontró el origen del déficit de fijación.

Esto llevó a Mittelstaedt a postular que los movimientos de la cabeza hechos en el seguimiento de una presa envuelve un sistema óptico retroalimentado. Una vez que la mantis asume el comportamiento de ataque, la asimetría de la imagen de la presa en el campo binocular de las componentes de los ojos debe de generar un mensaje nervioso que actúa en los músculos del cuello para que la asimetría sea reducida. Entonces, la asimetría de la imagen de la presa no es reducida a cero (excepto cuando ésta está directamente adelante), pero, como se dijo, este sistema es casi infalible, pudiéndose pensar que no está trabajando solo. Esto sugiere la presencia de otro sistema actuando constantemente como regulador que deberá operar en oposición al sistema óptico.

Los receptores en este segundo sistema pasan a ser los mecanoreceptores del cuello. Dos pares de placas cubiertas con pequeños cabellos están situados en cada uno de los lados de la base del cuello de la mantis (Figura F. IV. 2). Estas vellosidades están organizadas de tal forma para que cuando la cabeza esté situada simétricamente adelante de la base del cráneo, doble o presione cierta proporción de las vellosidades de ambos lados. Girando la cabeza hacia la derecha causa

mayor presión de los cabellos de la parte derecha mientras que aquéllos en la parte izquierda sufren obviamente menor deflexión, sucediendo lo mismo cuando se gira a la izquierda. Células sensitivas en la base de los cabellos son estimuladas en proporción a la deflexión. Experimentos posteriores en los que se aplican a los mecanoreceptores fuerzas extrañas muestran que la diferencial de las descargas nerviosas de las vellosidades del lado izquierdo y derecho constituyen una señal que genera, a través de los músculos del cuello, una fuerza de signo opuesto que tiende a desviar la cabeza. Cabe suponer que cuando la mantis gira su cabeza a una nueva posición la descarga del impulso nervioso conteniendo el comando de giro deberá exactamente balancear la señal contenida en la descarga diferencial de las ahora asimétricamente estimuladas placas de vellosidades. El modelo propuesto por Mittelstaedt del funcionamiento del sistema de la mantis en el seguimiento de una presa es el que se muestra en la figura F.IV.3.

Los sistemas retroalimentados óptico y propioceptivo operan a través de los músculos del cuello; el sistema propioceptivo está indicado en la figura rodeado por líneas discontinuas. La señal nerviosa Y_s actuando en los músculos del cuello es la diferencia de las señales provenientes del sistema óptico y los mecanoreceptores, y es expresado como la posición de la cabeza Y . La salida del sistema óptico X_s con la cabeza en esta posición

corresponde al déficit de fijación X, entonces Xs contiene información sobre el ángulo presa-cabeza así como del ángulo cuerpo-cabeza, la salida del sistema óptico puede en un momento dado activar la contracción del modelo de los músculos de las patas delanteras para que las proyecte hacia la presa en combinación con la recepción de un comando de ataque.

IV.3. SIMULACION DEL MODELO MATEMATICO.

El modelo utilizado en la simulación del ataque de una presa por la mantis religiosa es el que se muestra en la figura F.IV.4, que es el obtenido por Mittelstaedt con algunas modificaciones, entre las cuales podemos mencionar que existe una mayor simplicidad en la función que define el sistema de control sin perder las propiedades que éste muestra, además de facilitar su simulación con esta simplificación hecha puesto que la ecuación diferencial que define a la función presenta una mayor sencillez en su solución. Se puede observar que el ángulo de posición de la presa Z activa un sistema de control visual retroalimentado obteniéndose una posición de la cabeza Y. Este sistema de control busca centrar binocularmente la posición de la presa; sin embargo este sistema está afectado por otro sistema de realimentación constituido por los mecanoreceptores que al activarse generan una

senal que impide que la M.R. pueda centrar totalmente a la presa. El ángulo de la cabeza es variable con el tiempo y está representado por dY/dt y está dado como la diferencia del error del sistema (E) y el error producido por los mecanoreceptores (EM) para una posición de la cabeza en el tiempo. El valor E es la diferencia de la entrada Z y la posición de la cabeza Y en un momento t; EM es el error producido por los mecanoreceptores que, de acuerdo al modelo de Mittelstaedt, es el 15 % del ángulo de la cabeza en el mismo momento t. El valor dY/dt es la entrada del bloque H, que no es más que un integrador, obteniendo a la salida de éste el valor del ángulo de la posición de la cabeza Y.

Una comprensión más clara de este modelo la tenemos si obtenemos su representación en el dominio del tiempo; de esta forma tenemos que :

$$dY/dt = Y(0) + (E-EM) \quad \dots \quad (EIV.1)$$

$$E = Z - Y \quad \dots \quad (EIV.2)$$

$$EM = 0.15*Y \quad \dots \quad (EIV.3)$$

substituyendo EIV.2 y EIV.3 en EIV.1 :

$$dY/dt = Y(0) + Z - Y - 0.15Y = (Z - (1.15 Y)) \quad \dots \quad (EIV.4)$$

obteniendo la transformada de Laplace de EIV.4

$$sY(s) = Y(0) + (Z(s) - 1.15 Y(s))$$

$$sY(s) + 1.15 Y(s) = Y(0) + Z(s)$$

$$Y(s) (s + 1.15) = Y(0) + Z(s)$$

$$Y(s) = (Y(0) + Z(s)) / (s + 1.15)$$

$$Y(s) = Y(0)/(s+1.15) + Z(s)/(s+1.15) \dots \quad (\text{EIV.5})$$

si $Z = \text{cte.} = u/s$ y pasamos al dominio del tiempo a EIV.5'

$$Y(t) = Y(0) * \text{EXP}(-1.15t) + (u/1.15) * (1 - \text{EXP}(-1.15t)) \dots \quad (\text{EIV.6})$$

Observamos en EIV.6 que cuando t tiende a infinito las exponenciales son cero y por tanto la posición de la cabeza $Y(t)$ es igual $u/1.15$ lo cual nos indica que $Y(t)$ es el 85 % del valor u que es la posición de la presa. Además en la ecuación E.IV.3 el valor EM es el correspondiente X del modelo de Mittelstaedt que indica el ángulo con respecto a la cabeza con que deben ser orientadas las patas delanteras para realizar un ataque exitoso.

En el Programa de simulación, del cual se presenta un listado en el Apendice C, se considerará una mantis religiosa en el piso respondiendo al estímulo de una presa que tiene movimiento aleatorio al mismo nivel (puede ser considerada como un gusano), obteniéndose para cada una de las posiciones de la presa: 1) el ángulo de la presa Z con respecto al eje del cuerpo de la mantis; 2) el ángulo de la cabeza Y referido igual que Z ; y 3) el valor angular del déficit de fijación EM que está referida a la cabeza de

la mantis.

El final de la simulación es considerado cuando la presa está al alcance de la mantis y ésta es atacada con las patas delanteras con el ángulo EM.

Las funciones del programa pueden ser vistas en la Figura F.IV.5 que muestra su diagrama de flujo. La posición inicial de la mantis y la presa es obtenida en forma alatoria, necesitándose para esto obtener las coordenadas en X e Y para ambos; estas coordenadas son obtenidas por medio de una función generadora de números aleatorios que toma como valor inicial o "semilla" la hora, los minutos y segundos de la computadora garantizándose con esto una situación diferente en cada simulación.

Las coordenadas para el movimiento de la presa son calculados en base a una función que genera numeros aleatorios y de acuerdo al valor generado se escoge una de las ocho opciones de movimiento: 1)arriba; 2)arriba-derecha; 3)derecha; 4)abajo-derecha; 5)abajo; 6)abajo-izquierda; 7)izquierda; y (8)izquierda-arriba.

La solución numérica de la ecuación diferencial se realiza aplicando el método iterativo de Euler, el cual presenta las características de facilidad de implementación y obtención, principalmente, de las características de una solución CUALITATIVA de la ecuación, siendo esto uno de nuestros objetivos. Adicionalmente, controlamos el tamaño del incremento de

tiempo considerado, de tal manera que no se incurra en errores grandes en la solución obtenida, por lo que el incremento de tiempo utilizado es de 0.05 segs.

Para mayor información sobre cada uno de los módulos observar el listado del programa fuente ya que éste está totalmente documentado.

Los resultados de la simulación pueden ser obtenidos en dos formas:

a) Gráfica. Muestra en un monitor de alta resolución de color en graficación animada (con movimiento) de dos dimensiones la escena de simulación desde una vista superior(planta) señalando los diferentes ángulos Z,Y,X y el despliegue de las patas delanteras cuando la presa es atacada.

b) Numérica. Muestra un listado con la siguiente información: las coordenadas de la mantis; las coordenadas de la presa en cada uno de los puntos del movimiento y su correspondiente ángulo Z; asociado a cada Z se muestran los diferentes valores de Y y EM correspondientes al movimiento gradual de la cabeza en el alcance del ángulo de la presa que a su vez son producto de la solución numérica de la ecuación E.IV.4; valor de la distancia a la que fue atacada la presa.

La opción b es solamente usada cuando la computadora no cuenta con el equipo de graficación especificado o se

quiere obtener un reporte de la secuencia de eventos ocurridos durante la simulación.

La ejecución del programa se lleva a cabo sólo tecleando su nombre, es decir: SIMATAQ .

Aquí no se presentan fotografías de la simulación gráfica de esta conducta, ya que como sólo hay movimiento de la cabeza del animal y una sucesión de posiciones para la mosca, sería difícil seguir su evolución en una gráfica y llevaría demasiadas fotografías si se mostrara por cuadros.

A continuación se presentan los resultados de una simulación llevada a cabo con este programa con la opción (b) de salida.

 ---- SIMULACION ATAQUE A PRESA POR LA M.R. ----

MANTIS RELIGIOSA.

Posición : X=0 Y=0
 Long. Patas: 47

Presa: X= 30 Y= 120 ANG. = 76.0

ANG. CAB.	ANG. PAT.	SUMA
22.78912354	0.00000000	22.78912354
38.74151230	3.418368340	42.15988159
48.88267517	5.811226368	54.69390106
55.26362991	7.332400799	62.59603119
59.27394485	8.289544106	67.56349182
61.79402542	8.891091347	70.68511963
63.37761688	9.269103050	72.64672089
64.37272644	9.506642342	73.87937164
64.99803925	9.655908585	74.65394592
65.39098358	9.749705315	75.14068604
65.63790131	9.808647156	75.44654846
65.79306030	9.845685005	75.63874817
65.89056396	9.868958473	75.75952148
65.95183563	9.883584023	75.83541870
65.99033356	9.892774582	75.88311005
66.01452637	9.898549080	75.91307831
66.02973175	9.902178764	75.93190765
66.03928375	9.904459000	75.94374084
66.04528809	9.905892372	75.95117950
66.04905701	9.906792641	75.95584369
66.05142975	9.907358170	75.95878601
66.05291748	9.907713890	75.96063232
66.05385590	9.907937050	75.96179199
66.05444336	9.908078194	75.96252441
66.05480957	9.908165932	75.96297455
66.05504608	9.908221245	75.96326447
66.05519104	9.908256531	75.96344757
66.05528259	9.908278465	75.96356201
66.05534363	9.908291817	75.96363831
66.05538177	9.908301353	75.96368408
66.05540466	9.908307076	75.96371460
66.05541992	9.908309937	75.96372986
66.05542755	9.908312798	75.96373749
66.05542755	9.908313751	75.96374512
66.05542755	9.908313751	75.96374512

Presa: X= .30 Y= 80 ANG. = 69.4

ANG. CAB.	ANG. PAT.	SUMA
64.09948730	9.908313751	74.00780487
62.73033142	9.614922523	72.34525299
61.85993958	9.409548759	71.26948547
61.31227875	9.278990746	70.59127045
60.96808243	9.196841240	70.16492462
60.75178909	9.145212173	69.89700317
60.61587143	9.112768173	69.72863770
60.53046417	9.092380524	69.62284851
60.47679520	9.079568863	69.55636597
60.44306946	9.071518898	69.51458740
60.42187881	9.066459656	69.48834229
60.40856171	9.063281059	69.47183990
60.40019226	9.061284065	69.46147919
60.39493179	9.060028076	69.45496368
60.39162827	9.059239388	69.45086670
60.38955307	9.058743477	69.44829559
60.38824844	9.058432579	69.44667816
60.38742828	9.058237076	69.44566345
60.38691930	9.058114052	69.44503021
60.38658905	9.058036804	69.44462585
60.38638687	9.057988167	69.44437408
60.38625717	9.057957649	69.44421387
60.38617706	9.057937622	69.44411463
60.38612747	9.057926178	69.44405365
60.38609695	9.057918549	69.44401550
60.38607788	9.057913780	69.44399261
60.38606262	9.057910919	69.44397736
60.38605499	9.057909012	69.44396210
60.38605118	9.057908058	69.44396210
60.38604736	9.057907104	69.44395447
60.38604355	9.057906151	69.44394684
60.38604355	9.057906151	69.44394684
60.38604355	9.057906151	69.44394684
60.38604355	9.057906151	69.44394684

Presa: X= 30 Y= 40 ANG. = 53.1

ANG. CAB.	ANG. PAT.	SUMA
55.49188995	9.057906151	64.54979706
52.06598282	8.323782921	60.38976669
49.88808441	7.809896946	57.69798279
48.51772308	7.483211994	56.00093460
47.65647507	7.277657986	54.93413162
47.11526489	7.148470879	54.26373672
46.77517700	7.067289352	53.84246826
46.56147003	7.016276360	53.57774734
46.42717743	6.984220028	53.41139603
46.34278870	6.964076042	53.30686569

46. 28976059	6. 951417923	53. 24118042
46. 25643921	6. 943463802	53. 19990158
46. 23550034	6. 938465595	53. 17396545
46. 22234344	6. 935324669	53. 15766907
46. 21407318	6. 933351040	53. 14742279
46. 20887756	6. 932110786	53. 14098740
46. 20561218	6. 931331158	53. 13694382
46. 20355988	6. 930841446	53. 13440323
46. 20227051	6. 930533409	53. 13280487
46. 20146179	6. 930340290	53. 13180161
46. 20095062	6. 930218697	53. 13116837
46. 20063019	6. 930142403	53. 13077164
46. 20042801	6. 930094242	53. 13052368
46. 20030212	6. 930063725	53. 13036728
46. 20022202	6. 930045128	53. 13026810
46. 20017242	6. 930032730	53. 13020706
46. 20014191	6. 930025578	53. 13016891
46. 20012283	6. 930020809	53. 13014221
46. 20011139	6. 930017948	53. 13013077
46. 20010376	6. 930016518	53. 13011932
46. 20009995	6. 930015087	53. 13011551
46. 20009613	6. 930014610	53. 13011169
46. 20009232	6. 930014133	53. 13010788
46. 20009232	6. 930013657	53. 13010788
46. 20009232	6. 930013657	53. 13010788

Presa: X= 30 Y= 0 ANG. = 0.0

ANG. CAB.	ANG. PAT.	SUMA
30. 26106071	6. 930013657	37. 19107437
19. 10373878	4. 539158821	23. 64289956
12. 01086998	2. 865560532	14. 87643051
7. 547941208	1. 801630378	9. 349571228
4. 743070126	1. 132191062	5. 875261307
2. 980491877	. 7114604712	3. 691952467
1. 872906208	. 4470737576	2. 319979906
1. 176912308	. 2809359133	1. 457848191
. 7395578623	. 1765368283	. 9160947204
. 4647294581	. 1109336764	. 5756631494
. 2920305431	. 6970941275E-01	. 3617399633
. 1835085601	. 4380457848E-01	. 2273131460
. 1153146178	. 2752628177E-01	. 1428409070
. 7246235013E-01	. 1729719155E-01	. 8975954354E-01
. 4553448781E-01	. 1086935215E-01	. 5640384182E-01
. 2861333638E-01	. 6830172613E-02	. 3544351086E-01
. 1798028499E-01	. 4292000085E-02	. 2227228507E-01
. 1129859965E-01	. 2697042655E-02	. 1399564277E-01
. 7099906914E-02	. 1694789855E-02	. 8794697002E-02
. 4461498000E-02	. 1064985991E-02	. 5526483990E-02
. 2803552896E-02	. 6692246534E-03	. 3472777549E-02
. 1761719701E-02	. 4205329169E-03	. 2182252705E-02
. 1107043936E-02	. 2642579493E-03	. 1371301943E-02
. 6956533762E-03	. 1660565758E-03	. 8617099375E-03

.4371404066E-03	.1043480006E-03	.5414884072E-03
.2746939135E-03	.6557105371E-04	.3402649600E-03
.1726144401E-03	.4120408630E-04	.2138185228E-03
.1084688847E-03	.2589216456E-04	.1343610493E-03
.6816057430E-04	.1627033089E-04	.8443090337E-04
.4283130693E-04	.1022408560E-04	.5305539162E-04
.2691469126E-04	.6424695584E-05	.3333938730E-04
.1691287616E-04	.4037203325E-05	.2095007949E-04
.1062785304E-04	.2536931333E-05	.1316478483E-04
.6678417776E-05	.1594177888E-05	.8272595551E-05
.4196639566E-05	.1001762598E-05	.5198402050E-05

*** CAPTURA DE PRESA *** Distancia = 30

 ---- FIN SIMULACION ----

En el ejemplo de la simulación mostrado se observa que la M.R. tiene una posición de $x=0$ e $y=0$, permaneciendo en esta posición hasta el fin de la simulación: el alcance de las patas delanteras es de 47 unidades.

La posición inicial de la presa es de $x=30$ e $y=120$, dando como resultado de esta posición un ángulo referido al eje X de 76 grados.

Una vez que se tiene una posición de presa, se aplica el modelo, observándose en qué forma los valores de ángulo de la cabeza de la M.R. van incrementándose desde un valor de 22.78 grados hasta llegar a 66.05 grados, a la par que el ángulo que debe ser sumado al obtenido visualmente debido al sistema propioceptivo y músculos del cuello, va incrementándose desde su valor

inicial de 0.00 hasta 9.90 grados; se observa que al sumar los valores finales del ángulo de la cabeza y el ángulo de las patas se obtiene un valor de 75.96 grados, que es el ángulo de la presa; en el momento de obtener este valor la mantis tiene localizada la presa y está lista para atacarla, solamente que la distancia que los separa es mayor que el alcance de sus patas.

Después de esto se observa un movimiento de la presa, obteniéndose posiciones de $x=30$ e $y=80$, teniendo esta posición un ángulo de 69.4 grados; con estos valores se vuelve a activar el modelo de localización de presa y se ve la evolución paulatina que tienen el ángulo de la cabeza y el de las patas desde su posición anterior (66.05 y 9.90 grados respectivamente), hasta llegar a su posición final de 60.38 y 9.05, que si se suman se obtiene un ángulo de 64.4 grados, que es la posición de la presa; en este momento, una vez localizada la presa se intenta atraparla, pero la distancia no es adecuada todavía.

Posteriormente la presa realiza otro movimiento obteniéndose las coordenadas de $x=30$ e $y=40$, obteniéndose con esto un ángulo de 53.1 grados; con estas posiciones la M.R. desde sus posiciones anteriores de ángulo de cabeza y patas va dirigiéndose hacia la posición de la presa, logrando localizarla con valores de 46.2 y 6.93 grados respectivamente, pero sin estar aún a su alcance la presa para atraparla.

Finalmente la presa realiza otro movimiento, dándose su posición por las coordenadas $x=30$ e $y=0$, obteniéndose con esto un ángulo de 0.00 y una distancia entre presa-mantis de 30 ; la mantis localiza a la presa con esta nueva posición con ángulos de cabeza y patas de 0.0000041 y $.000001$ grados, evaluando posteriormente a esto la distancia que la separa de la presa, y dado que ésta es menor que el alcance de sus patas, activa el comando de atrapa presa, obteniéndose con esto el fin de la simulación.

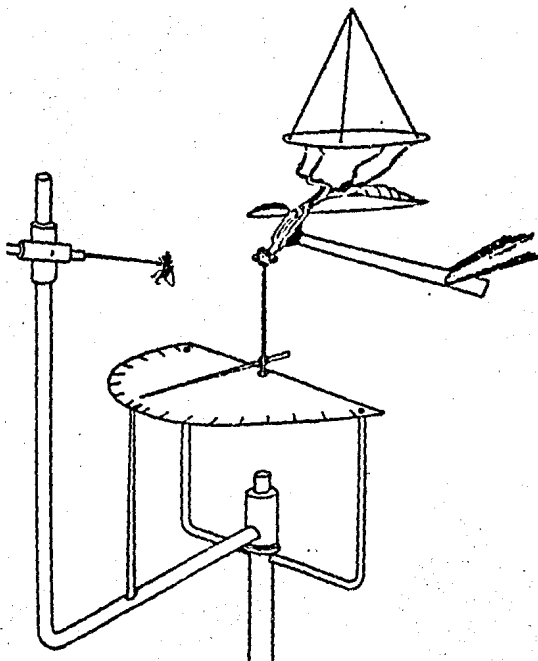


Figura F.IV.1. Dispositivo para la medición del dificit de fijación durante la localización de presa en la mantis religiosa. La mantis es fijada en el protorax. La cabeza esta libre para rotar a través de los ejes verticales, su posición es mostrada por un apuntador de madera de balsa. La presa puede ser presentada a diferentes angulos, y los angulos hechos por el movimiento de la cabeza y presa son medidos en la misma escala.

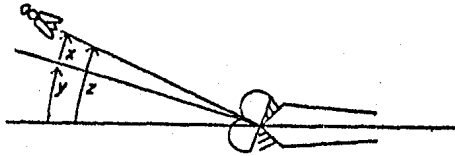


Figura F.IV.2. Cabeza y protórax de una mantis fijas en el seguimiento de una presa: X, déficit de fijación; y, desviación de la cabeza del eje del cuerpo cuando la presa está fija; Z, desviación de la presa del eje del cuerpo.

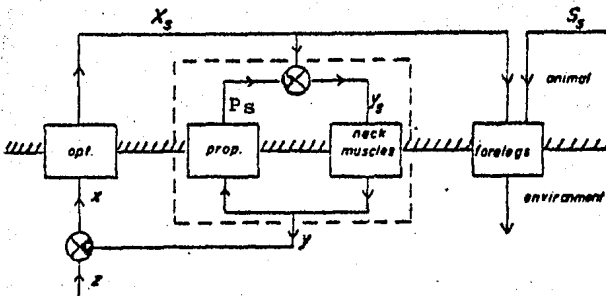


Figura F.IV.3. Modelo de control para la localización de una presa. - Cuadrados representan elementos nerviosos y musculares de cada subsistema. El sistema óptico retroalimentado contiene dentro de sí el sistema propioceptor (líneas discontinuas) que corre contrario a él. La salida óptica X_s contiene la señal de puntería correcta que toma efecto una vez que se activa la señal de atrapar S_s .-- Los siguientes en blanco de los círculos, indican adición; los negros, sustracción...

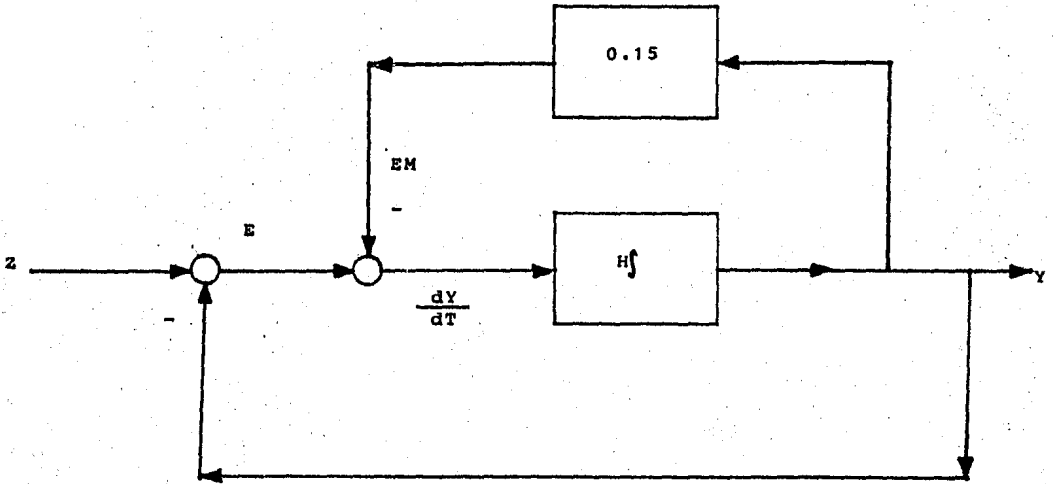


Figura F.IV.4. Modelo de Control para definir la conducta de ataque de presa de la Mantis Religiosa.

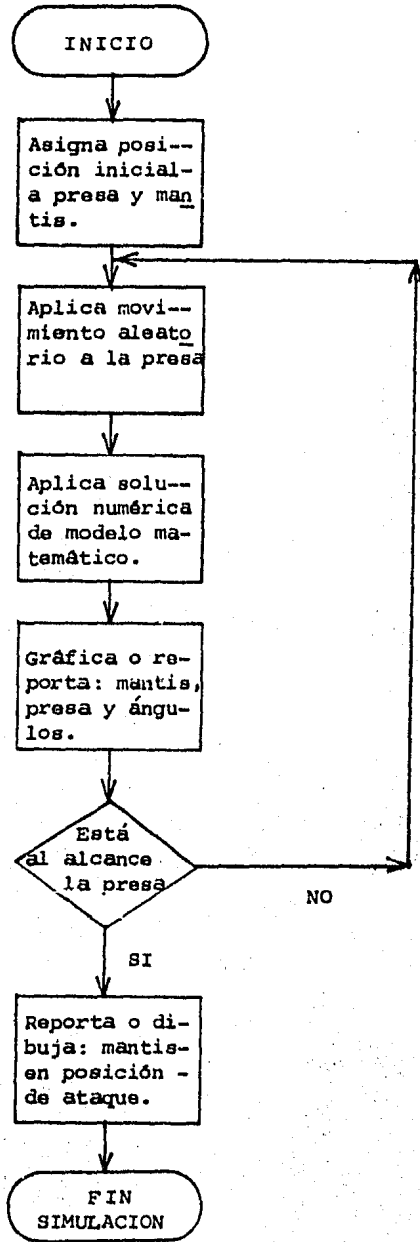


Figura F.IV.5. Diagrama de bloques del programa de simulación de ataque de una presa de la mantis religiosa.

C A P I T U L O V
C O N C L U S I O N E S .

En el deseo de profundizar en el conocimiento del funcionamiento del sistema nervioso, siguiendo el enfoque de asociación de mecanismos neuronales con el control de ciertas conductas estereotipadas en los animales (neuroetología), se eligió a la Mantis Religiosa, un insecto que presenta una serie de ventajas (ver introducción) para estudios de este tipo.

Adicionalmente, existe una carencia de estructuración, organización y planeación en los estudios empíricos realizados sobre este animal, amén de una no muy nutrida cantidad de estudios etológicos avocados solamente a dos de sus conductas: la captura de presa y la respuesta deimática ante un depredador, además de algunos estudios fisiológicos y anatómicos.

En el presente trabajo hemos estudiado como la Mantis Religiosa interactúa con su medio ambiente en función de los tipos y las dimensiones de los estímulos que están presentes en su campo visual, las condiciones métricas de los estímulos en función de su cuerpo y los objetivos que desencadenan. Este trabajo claramente muestra que estos animales definen sus pautas motoras con sus distintas modalidades en función de una percepción precisa de sus tres dimensiones; estos animales además pueden planear la ruta a seguir en base a la evaluación

precisa de sus tres dimensiones buscando llegar al objetivo por medio de tácticas específicas.

Los logros obtenidos en esta tesis, pueden ser agrupados de la siguiente forma:

1. Hemos propuesto un modelo global de coordinación visuomotora en estos animales que nos permite reproducir su conducta ante diferentes circunstancias y que nos permite postular el tipo de procesamiento de información que realiza su sistema nervioso.

2. El desarrollo e implementación en computadora digital de modelos teóricos(matemáticos) de las conductas de búsqueda de refugio y salto balístico nos permitió hacer predicciones del comportamiento del animal ante diferentes situaciones, y hacer extrapolaciones de comportamiento para estadios diferentes.

3. La realización de simulaciones de los modelos matemáticos obtenidos, a través de medios gráficos, permite su seguimiento visual, mostrándose así la evolución del comportamiento que definen.

4. La simplificación del modelo propuesto por Mittelstaedt[05] para el ataque de una presa a uno más sencillo y que involucra todas las funciones del anterior nos permitió el llevar a cabo su simulación gráfica y

numérica en una forma más sencilla en cuanto a su implementación

Los puntos 1 y 2 son de extrema importancia ya que nos permiten postular nuevas hipótesis, que pueden ser analizadas por medio de estudios fisiológicos y etológicos, sobre el tipo de respuestas que debemos encontrar en cada una de las circunstancias analizadas, como por ejemplo, las diferentes condiciones métricas de zanja y objetivos durante la conducta de búsqueda de refugio.

Es importante añadir también que la teoría de esquemas propuesta en el presente trabajo no es exclusiva para la Mantis Religiosa, sino que se puede aplicar para cualquier animal con una conducta sensorimotora clara. Asimismo, esta teoría podría fácilmente ser aplicada al diseño de robots que tienen que moverse en un espacio tridimensional con diversos objetos y donde tienen que alcanzar ciertos objetivos.

A P E N D I C E A.

P R O G R A M A D E S I M U L A C I O N D E B U S Q U E D A
D E R E F U G I O D E L A M. R.

```

C*****
C****                                     ****
C****          P R O G R A M A   D E F I N E          ****
C****                                     ****
C*****
C
C  Simulacion de la Mantis Religiosa.(PARTE 1: DEFINICION DEL ESPACIO)
C
C  Vicente Alcantara Guerrero
C
C  19-ene-1984
C
C  Bibliotecas Ligadas:
C
C  FGRAF1
C  GRAF2
C  MANLIB1
C  CRTLIB
C  FORLIB
C
C
C  program define
C  byte cmd
C  byte comand(7),sub1(5),sub2(3)
C  byte logocm(22),logocu(24)
C  byte logobs(32)
C  integer xilogo,x2logo,yilogo,y2logo
C  integer x,y,z
C  integer xmin,xmax,zmin,zmax,ymin,ymax
C  integer xpaso,ypaso,zpaso
C  integer xman,yman,zman
C  integer resolx,resolz
C  integer objeto(40)
C  logical imagen
C  byte file(11)
C  common/proy/psi
C  common/increm/xpaso,ypaso,zpaso
C  common/maxim/xmax,ymax,zmax
C  common/minim/xmin,ymin,zmin
C  common/valor/pi
C  common/comun1/l,j,k,1
C  common/comun2/n,11,12,13
C  common/infman/xman,yman,zman,theta,imagen
C  common/infobj/objeto,nobj
C  common/valman/scal,largo,impaso
C  data ncmd/6/
C  data nsub1/5/
C  data nsub2/3/
C  data noblim/8/
C  data logocm/'C','O','M','A','N','D','O',' ','',
*      'M','a','n','t','i','s',' ',
*      'O','b','j','e','t','o','@'/
C  data logocu/'M','O','V','I','C','E','R','O','R',' ',
*      '<','>','@'/
*      '<','>','@'/

```



```

C LARGO : Largo del Cuerpo de la Mantis Religiosa( en X)
C
C IMPASO : Longitud del Paso de la Mantis Religiosa.
C
C X,Y,Z : Coordenadas del punto del cursor en el espacio tridimensional
C
C
C     call grafix
C     call init1
C     call estrng('@')
C     pi=atan2(0,-1)
C     degr=pi/180.
C     largo=largo*scal
C     impaso=xmpaso*scal
C
C     psi=psi*degrad
C
C     theta=theta*degrad
C Define los valores de Xmax y Zmax en funcion de los Asignados a Y
C
C     xmax=reso1x-ifix(cos(psi))*ymax)
C     zmax=reso1z-ifix(sin(psi))*ymax)
C
C     x=xmin
C     y=ymin
C     z=zmin
C     call square
C Definira los colores a usar
C
C     call defclr(5,0,15,0)
C     call defclr(7,0, 8,8)
C     call defclr(12,15,0,0)
C -----
C OBTENCION DE COMANDOS
C -----
10     continue
C     call logo(x1logo,y1logo,logocu)
C     call logo(x2logo,y2logo,logocm)
15     call flash(x,y,z)
C     call inkey(cmd)
C     if(cmd .eq. 0)go to 15
C     do 20 icmd=1,ncmd
C     if(cmd .eq.comand[icmd])go to 30
20     continue
C
C error: comando no valido
C
C
25     call ring
C     go to 15
30     continue
C     go to (600,600,600,600,100,200), icmd
C
C -----
C -- DEFINE MANTIS RELIGIOSA

```



```

C -----
100  continue
      if[nobj.ne.0]go to 102
      call ponerr(x2logo,y2logo,'ERROR:No hay definido objeto alguno@'
*      ,itime)
      go to 10
102  call logo(x2logo,y2logo,'MANTIS:L para asignar posicion@')
105  call lnkey(cmd)
      call flash(x,y,z)
      if(cmd .eq.0)go to 105
      do 106 icmd=1,nsub1
        if(cmd .eq. sub1(icmd))go to 108
106  continue
      call ring
      go to 105
108  go to(110,110,110,110,115), icmd
110  call muevec(x,y,z, icmd)
      go to 105
115  continue
      i1=x+largo
      i2=y+1
      call checa2(objeto,nobj,x,y,i1,i2,icode)
      if(icode.eq.0)go to 120
      call ponerr(x2logo,y2logo,'ERROR:Interseccion con Objeto@',
*      itime)
      go to 100
120  continue
      xman=x
      yman=y
      zman=z
      call drwman
C se inicia la Simulacion.
      go to 300

```

C
C -----
C -- DEFINE OBJETO
C -----

```

200  continue
      if[nobj.ne.noblim]go to 204
      call ponerr(x2logo,y2logo,'ERROR:limite de objetos definidos@',
*      itime)
C deshabilita posibilidad de hacer otra definicion de Objeto
      logocm(15)='@'
      ncmd=ncmd-1
      go to 10
204  Nobj=Nobj+1
      iofset=(Nobj-1)*5

```

C
C recibimos primer y segundo extremos de la base
C

```

201  ndim=0
202  continue
      if[ndim.eq.0]logobs(31)='1'
      if[ndim.eq.1]logobs(31)='2'
      call logo(x2logo,y2logo,logobs)

```

```

205  continue
      call flash(x,y,z)
      call inkey(cmd)
      if(cmd.eq.0)go to 205
      do 206 icmd=1,nsubl
        if(cmd .eq. sub1(icmd))go to 208
206  continue
      call ring
      go to 205
208  go to(210,210,210,210,215),icmd
210  continue
      call muevec(x,y,z,icmd)
      go to 205
215  continue
      call checal(objeto,nobj-1,x,y,icode,ii)
      if(icode.eq.1)go to 217
      call ponerr(x2logo,y2logo,'ERROR:punto dentro de otro objeto@',
        itime)
      go to 202
217  continue
C  checar si se definio correctamente una base(una area no una linea)
      if(ndim.eq.0)go to 214
      if(.not.(x.eq.objeto(indx).or.y.eq.objeto(indx+1)))go to 214
      call ponerr(x2logo,y2logo,'ERROR:mala definicion de base@',
        itime)
      go to 202
214  ndim=ndim+1
      indx=iofset+ndim*2-1
      objeto(indx)=x
      objeto(indx+1)=y
      call d32(x,y,z,i,j)
      call xdot(i,j,ibsc1r)
      if(ndim .lt. 2)go to 202
C  pone en la primer coordenada la esquina que tiene el menor valor en x e y
C  y en la segunda la que tiene mayor x e y
      indx=iofset+1
      do 219 i=1,2
        l=indx+2
        if(objeto(indx).lt.objeto(l))go to 218
        k=objeto(indx)
        objeto(indx)=objeto(l)
        objeto(l)=k
218  indx=indx+1
219  continue
      call drwobj(objeto(iofset+1),1,ibsc1r)
      call checa2(objeto,nobj-1,objeto(iofset+1),objeto(iofset+2),
        objeto(iofset+3),objeto(iofset+4),icode)
      if(icode.eq.0)go to 220
      call ponerr(x2logo,y2logo,'ERROR:interseccion de objetos@',
        itime)
      call drwobj(objeto(iofset+1),1,iclear)
      call drwobj(objeto,nobj-1,iobc1r)
      go to 201
C
C  definicion de la altura del Objeto

```

```

C
220  continue
      call logo(x1logo,y1logo, 'CURSOR: ^ , !@')
      call logo(x1logo,y1logo-10, ' | VE')
      call logo(x2logo,y2logo, 'OBJETO:L al definir altura@')
225  continue
      call flash(x,y,z)
      call inkey(cmd)
      if(cmd.eq.0)go to 225
      do 226 icmd=1,nsub2
        if(cmd.eq.sub2(icmd))go to 228
226  continue
      call ring
      go to 225
228  continue
      go to(230,230,235),icmd
230  continue
      icmd=icmd+4
      call muevec(x,y,z,icmd)
      go to 225
235  continue
C  checara si se definio Altura del objeto
      if(z.gt.zmin)go to 238
      call ponerr(x2logo,y2logo,'ERROR:No se definio Altura@',itime)
      go to 220
238  ndim=ndim+1
      indx=iodefset+ndim*2-1
      objeto(indx)=z
      z=zmin
      call drwobj(objeto,nobj,iobclr)
      call logo(x1logo,y1logo-10, '@')
      go to 10

```

```

C
C -----
C -- PRINCIPIA SIMULACION
C -----

```

```

300  continue
      call logo(x1logo,y1logo,'@')
      call logo(x2logo,y2logo,'PRINCIPIA SIMULACION@')
      call open(ifile,file,0)
      write(ifile,310)Nobj
310  format(' Numero de Objetos           ',15)
      do 315 i=1,nobj
        k=(i-1)*5+1
        l=k+4
        write(ifile,320)i,(objeto(j),j=k,l)
315  continue
320  format(' Objeto no. ',i2,'           ',515)
      write(ifile,325)xman,yman,zman
325  format(' Coordenadas de la Mantis     ',315)
      write(ifile,330)scal
330  format(' Escala Mantis                   ',75.2)
      write(ifile,335)largo
335  format(' Largo del cuerpo Mantis          ',15)
      write(ifile,340)impaso

```

```

340  format(' Longitud del Paso Mantis      ',15)
      endfile ifile
      go to 1000

C
C -----
C --  MOVIMIENTO DE CURSOR
C -----
600  continue
      call muevec(x,y,z,icmd)
      go to 15

C
C
1000 continue
      end

C
C
C*****
C**  subroutine pinta espacio tridimensional  **
C*****
C
      subroutine square
      common/increm/ixp,iyp,izp
      common/maxim/ixmax,iymax,izmax
      common/minim/ixmin,iymin,izmin
      common/comun1/i11,jj1,i12,jj2
      common/comun2/ix,iy,iz,ij

C
C
      icolor=15
      iy=iymin+1
C  hara la cuadrícula del plano X-Y
      do 10 i11=ixmin,ixmax,ixp
        i12=i11+ixp
        do 10 i=iy,iymax,iyp
          jj1=i-1
          jj2=jj1+iyp
          call d32(i11,jj1,izmin,i1,j1)
          call d32(i11,jj2,izmin,i2,j2)
          call d32(i12,jj2,izmin,i3,j3)
          call d32(i12,jj1,izmin,i4,j4)
          call xline(i1,j1,i2,j2,icolor)
          call xline(i2,j2,i3,j3,icolor)
          call xline(i3,j3,i4,j4,icolor)
          call xline(i4,j4,i1,j1,icolor)
10      continue
C  hara el rayado del plano X-Z
C
      do 30 i=izmin,izmax,izp
        call d32(ixmin,iymax,i,i1,j1)
        call d32(ixmax,iymax,i,i2,j2)
        call xline(i1,j1,i2,j2,icolor)
30      continue
C  hara el rayado del plano Y-Z
C
      do 40 i=izmin,izmax,izp

```

```

        call d32(ixmin,iymin,i,i1,j1)
        call d32(ixmin,iymax,i,i2,j2)
        call xline(i1,j1,i2,j2,icolor)
40    continue
        return
        end
C
C
C*****
C**  subroutine dibuja objeto          **
C*****
C
        subroutine drwobj(objeto,N,icolor)
        integer objeto(1)
        integer zmin
        common/minim/ix,iy,zmin
        common/comun1/ixmin,iymin,ixmax,iymax
        common/comun2/idwn1,jdwn1,idwn2,jdwn2
        if(n.eq.0)go to 500
        do 10 i=1,n
        iof=(i-1)*5
        ixmin=objeto(iof+1)
        iymin=objeto(iof+2)
        ixmax=objeto(iof+3)
        iymax=objeto(iof+4)
        ialt=objeto(iof+5)
C parte inferior
        call d32(ixmin,iymin,zmin,idwn1,jdwn1)
        call d32(ixmin,iymax,zmin,idwn2,jdwn2)
        call d32(ixmax,iymax,zmin,idwn3,jdwn3)
        call d32(ixmax,iymin,zmin,idwn4,jdwn4)
C parte superior
        call d32(ixmin,iymin,ialt,iup1,jup1)
        call d32(ixmin,iymax,ialt,iup2,jup2)
        call d32(ixmax,iymax,ialt,iup3,jup3)
        call d32(ixmax,iymin,ialt,iup4,jup4)
C
        call xline(idwn1,jdwn1,idwn2,jdwn2,icolor)
        call xline(idwn2,jdwn2,idwn3,jdwn3,icolor)
        call xline(idwn3,jdwn3,idwn4,jdwn4,icolor)
        call xline(idwn4,jdwn4,idwn1,jdwn1,icolor)
C
        call xline(iup1,jup1,iup2,jup2,icolor)
        call xline(iup2,jup2,iup3,jup3,icolor)
        call xline(iup3,jup3,iup4,jup4,icolor)
        call xline(iup4,jup4,iup1,jup1,icolor)
C
        call xline(idwn1,jdwn1,iup1,jup1,icolor)
        call xline(idwn2,jdwn2,iup2,jup2,icolor)
        call xline(idwn3,jdwn3,iup3,jup3,icolor)
        call xline(idwn4,jdwn4,iup4,jup4,icolor)
10    continue
500   continue
        return
        end

```

```
C
C
C*****
C**  subroutine "flash" cursor      **
C*****
```

```
C
    subroutine flash(i1,jj,z)
    integer x,y,z
    integer xpaso,ypaso,zpaso
    common/increm/xpaso,ypaso,zpaso
    common/comun1/i1,i2,i3,i4
    common/comun2/j1,j2,j3,j4
    icolor=12
    ix=i1+xpaso
    iy=jj+ypaso
    x=i1+2
    y=jj+2
    ix=ix-2
    iy=iy-2
    call d32(x,y,z,i1,j1)
    call d32(x,iy,z,i2,j2)
    call d32(ix,iy,z,i3,j3)
    call d32(ix,y,z,i4,j4)
```

```
C
    call xline(i1,j1,i2,j2,icolor)
    call xline(i2,j2,i3,j3,icolor)
    call xline(i3,j3,i4,j4,icolor)
    call xline(i4,j4,i1,j1,icolor)
```

```
    call delay(2000)
    icolor=0
    call xline(i1,j1,i2,j2,icolor)
    call xline(i2,j2,i3,j3,icolor)
    call xline(i3,j3,i4,j4,icolor)
    call xline(i4,j4,i1,j1,icolor)
    return
end
```

```
C
C
C*****
C**  subroutine pon logotipo en Pantalla  **
C*****
```

```
C
    subroutine logo(i,j,texto)
    byte texto(1)
    common/comun1/k,l,m,n
    call resbox(1,225,24,241)
    call hscale(2,755,1,482)
    k=i+375
    l=j+8
    n=j-3
    do 10 m=n,1
    call hxline(i,m,k,m,0)
    continue
    call hxttext(i,j,15,texto)
```

10

```

call scale(1,378,1,241)
return
end
C
C
C*****
C**  subroutine dibuja Mantis Religiosa.  **
C*****
C
subroutine drwman
integer xman,yman,zman
logical imagen
integer*1 tronco(43),cabeza(29),ojo(17)
integer*1 ptip1(25),ptdp1(09),pdip1(21),pddp1(13)
integer*1 ptip2(13),ptdp2(09),pdip2(17),pddp2(17)
integer*1 patfro(25)
common/infman/xman,yman,zman,theta,imagen
common/patman/itipl, idipl, iddp1, itdp1, itip2, idip2, iddp2, itdp2
C
C Datos de las diferentes partes del cuerpo de la Mantis Religiosa
C
C PTIPi pata trasera Izquierda en posicion i
C PDIPi pata delantera Izquierda en posicion i
C PDDPi pata delantera Derecha en posicion i
C PTDPi pata trasera Derecha en posicion i
C PATFRO patas Frontales
C
data tronco/40,
* -32, 15, 106, -35,
* 106, -35, 4, -4,
* 4, -4, 16, -22,
* 16, -22, 33, -24,
* 33, -24, 43, -32,
* 43, -32, 106, -35,
* 88, -35, 33, -24,
* 12, -21, 0, -3,
* 0, -3, -19, 5,
* -34, 8, -28, 5/
data cabeza/28,
* -32, 15, -34, 20,
* -34, 20, -39, 24,
* -39, 24, -45, 21,
* -45, 21, -45, 18,
* -45, 10, -43, 02,
* -43, 02, -36, 11,
* -28, 05, -32, 10/
data ojo/16,
* -40, 18, -41, 18,
* -41, 18, -41, 17,
* -41, 17, -40, 17,
* -40, 17, -40, 18/
C
C DEFINICION PATAS EN POSICION 1
C
data ptip1/24,

```

```

*      23, -21, 30, -32,
*      30, -32, 38, -36,
*      38, -36, 69, -62,
*      69, -62, 70, -70,
*      33, -24, 36, -27,
*      36, -27, 38, -36/
data pdipl/20.
*      12, -21, 12, -25,
*      12, -25, -18, -25,
*      -18, -25, -20, -60,
*      16, -22, 16, -28,
*      16, -28, -18, -25/
data pddpl/12,
*      10, -17, -13, -20,
*      -13, -20, -12, -50,
*      -12, -21, -13, -20/
data ptdpl/08,
*      46, -32, 83, -57,
*      83, -57, 85, -60/

```

C DEFINICION PATAS EN POSICION 2

```

C
data pddp2/16.
*      12, -21, 2, -22,
*      2, -22, -20, -15,
*      -20, -15, -33, -24,
*      10, -17, 2, -22/
data ptip2/12.
*      23, -21, 32, -45,
*      32, -45, 17, -73,
*      33, -24, 32, -45/
data pdip2/16,
*      12, -21, 6, -31,
*      0, -31, -31, -37,
*      -31, -37, -44, -49,
*      16, -22, 0, -31/
data ptdp2/08,
*      38, -26, 45, -45,
*      45, -45, 30, -60/

```

C patas frontales

```

C
data patfro/24.
*      -19, 05, -5, -7,
*      -5, -7, -10, -10,
*      -23, -12, -41, -10,
*      -41, -10, -41, -5,
*      -41, -5, -15, -5,
*      -28, 5, -15, -5/
data itip1, idipl, iddp1, itdp1, itip2, idip2, iddp2, itdp2
* / 1, 2, 3, 4, 8, 9, 10, 11/
data iclman/5/, iclojo/12/

```

C escalara el nuevo espacio para la mantis
call d32(xman, yman, zman, io, jo)


```

ix1=1-io
ix2=378-io
iy1=1-jo
iy2=241-jo
call scale(ix1,ix2,iy1,iy2)
C
C dibujara el Tronco
C
    call drwpar(tronco,theta,iclman,imagen)
C
C dibujara la cabeza
C
    call drwpar(cabeza,theta,iclman,imagen)
    call drwpar(ojo,theta,iclojo,imagen)
C
C dibujara las patas de la posicion 2
C
    call drwpar(pdip1,theta,idip1,imagen)
    call drwpar(pddp1,theta,iddp1,imagen)
    call drwpar(ptip1,theta,itip1,imagen)
    call drwpar(ptdp1,theta,itdp1,imagen)
C
C dibujara las patas de la posicion 1
C
    call drwpar(pdip2,theta,idip2,imagen)
    call drwpar(pddp2,theta,iddp2,imagen)
    call drwpar(ptip2,theta,itip2,imagen)
    call drwpar(ptdp2,theta,itdp2,imagen)
C
C dibujara las patas frontales
C
    call drwpar(patfro,theta,iclman,imagen)
C
C Hara los colores de las patas de posicion 2 negros,para que no se vean
C
    do 20 i=itip2,itdp2
        call defclr(1,0,0,0)
20    continue
C
C Hara los colores de las patas de la posicion 1 de color iclman(verde)
    do 10 i=itip1,itdp1
        call defclr(1,0,15,0)
10    continue
C
C regresa al espacio de Graficacion original
C
    call scale(1,378,1,241)
    return
    end
C
C
C*****
C**  subroutine dibuja parte del cuerpo de      **
C**      la Mantis Religiosa                    **
C*****

```

```

C
  subroutine drwpar(parte, theta, icolor, imagen)
  integer*1 parte(1)
  real theta
  logical imagen
  common/comun1/l,j,k,n
  common/comun2/ix1,iy1,ix2,iy2
  common/valman/scal,largo,impaso
  data ixorig,izorig/-50,-75/
  n=parte(1)
  n=n+1
do 10 i=2,n,4
  ix1=parte(i)-ixorig
  iy1=parte(i+1)-izorig
  ix2=parte(i+2)-ixorig
  iy2=parte(i+3)-izorig
  call rota(ifix(scal*ix1),ifix(scal*iy1),theta,i1,j1)
  call rota(ifix(scal*ix2),ifix(scal*iy2),theta,i2,j2)
  if(.not. imagen)go to 05
  call voltear(i1,j1,1)
  call voltear(i2,j2,1)
05  call xline(i1,j1,i2,j2,icolor)
10  continue
  return
  end

```

```

C*****
C**  subroutine obten espejo de Cuerpo de      **
C**      Mantis Religiosa                    **
C*****

```

```

C
  subroutine voltear(xo,yo,ieje)
  integer xo,yo
  go to(100,200),ieje
C espejo respecto al eje X
C
100  continue
  yo=-yo
  go to 300
C espejo respecto al eje Y
C
200  continue
  xo=-xo
  continue
300  return
  end

```

```

C*****
C**  subroutine checa si hay punto de un      **
C**      objeto dentro del volumen de otro   **
C*****

```

```

C
  subroutine checal(objeto,nobj,x,y,icode,iobjet)
  integer objeto(1),x,y
  common/comun1/l,j,k,l

```

C

```

C          |
C          | 1 si el punto no esta dentro de un volumen
C  icode= -1
C          | 2 si esta dentro de un volume e IOBJET es el volumen que
C          |         continue al punto
C
C          icode=1
C          if(nobj.eq.0)go to 500
C          do 10 i=1,nobj
C            indx=(i-1)*5+1
C            if(.not.(x.gt.objeto(indx).and.x.lt.objeto(indx+2)))go to 10
02         if(.not.(y.gt.objeto(indx+1).and.y.lt.objeto(indx+3)))go to 10
C            icode=2
C            iobjet=i
C            i=nobj
C          10 continue
C          500 continue
C          return
C          end
C*****
C**  subroutine checa si hay interseccion      **
C**  de los otros cuerpos con el de definicion **
C*****
C
C  subroutine checa2(objeto,nobj,x1,y1,x2,y2,icode)
C  integer objeto(1) x1,y1,x2,y2
C  integer x1,x2,y1,y2
C  logical isw(4),iswl(4)
C
C  common/comun1/i,j,indx,1
C
C  icode=0
C  if(nobj.eq.0)go to 500
C  do 10 i=1,nobj
C    do 5 j=1,4
C      isw(j) = .false.
C      iswl(j) = .false.
05  continue
C      indx=(i-1)*5+1
C      if(x1.gt.objeto(indx).and.x1.lt.objeto(indx+2))isw(1)=.true.
C      if(x2.gt.objeto(indx).and.x2.lt.objeto(indx+2))isw(2)=.true.
C      if(y1.gt.objeto(indx+1).and.y1.lt.objeto(indx+3))isw(3)=.true.
C      if(y2.gt.objeto(indx+1).and.y2.lt.objeto(indx+3))isw(4)=.true.
C
C      if(x1.le.objeto(indx)) iswl(1)=.true.
C      if(x2.ge.objeto(indx+2))iswl(2)=.true.
C      if(y1.le.objeto(indx+1))iswl(3)=.true.
C      if(y2.ge.objeto(indx+3))iswl(4)=.true.
C
C          Algunos esquina esta dentro de otro cuerpo?
C          if(isw(1).and.isw(3))icode=1
C          if(isw(1).and.isw(4))icode=1
C          if(isw(2).and.isw(3))icode=1
C          if(isw(2).and.isw(4))icode=1

```

```
C
C
C           Cuerpo conteniendo dos o mas esquinas de otro?
C           if((isw(1).or.isw(2)).and.(isw(3).and.isw(4)))icode=1
C           if((isw(3).or.isw(4)).and.(isw(1).and.isw(2)))icode=1
```

```
C
C           Un cuerpo contiene a Otro o es el mismo?
C           if[isw1(1).and.isw1(2).and.isw1(3).and.isw1(4)]icode=1
10  continue
500 continue
    return
    end
```

```
C
C
C*****
C**  subroutine pon mensaje de error en el      **
C**          monitor.                          **
C*****
C
```

```
C
    subroutine ponerr(ixlogo,iylogo,msg,itime)
    byte msg(1),cmd
10  continue
    call logo(ixlogo,iylogo,msg)
    call ring
    call delay(itime)
    call logo(ixlogo,iylogo.'Presione cualquier tecla@')
    call delay(itime)
    call inkey(cmd)
    if(cmd.eq.0)go to 10
    return
    end
```

```
C
C
C*****
C**  subroutine haz loop para time delay      **
C**          **                                **
C*****
C
```

```
C
    subroutine delay(n)
10  do 10 i=1,n
    continue
    return
    end
```

```

C*****
C****          P R O G R A M A   S I M U L A          ****
C****          ****
C*****
C
C Simulacion de la Mantis Religiosa. (PARTE 2: SIMULACION DEL ESPACIO)
C
C Vicente Alcantara Guerrero
C
C 26-ene-1984
C
C Bibliotecas Ligadas:
C
C   FGRAF1
C   GRAF2
C   DEFINEM
C   MANLIB1
C   CRTLIB
C   FORLIB
C
C   program simula
C   integer x1logo, x2logo, y1logo, y2logo
C   integer x, y, z
C   integer xmin, xmax, zmin, zmax, ymin, ymax
C   integer xpaso, ypaso, zpaso
C   integer xman, yman, zman
C   integer resolx, resolz
C   integer objeto(40)
C   logical imagen
C   byte file(11)
C
C   integer pagina
C   byte tecla, esc
C   integer haycha, indcha
C   integer hayobs, indobs
C
C   common/proy/psi
C   common/maxim/xmax, ymax, zmax
C   common/minim/xmin, ymin, zmin
C   common/valman/scal, largo, impaso
C   common/valor/pi
C   common/comun1/i, j, k, l
C   common/comun2/n, i1, i2, i3
C Informacion de Objetos y Mantis
C   common/infobj/objeto, nobj
C   common/infman/xman, yman, zman, theta, imagen
C   common/logot/x1logo, y1logo, x2logo, y2logo
C
C   data x1logo, y1logo/4.460/, x2logo, y2logo/378, 460/
C   data resolx, resolz/350, 225/
C   data psi/45./, theta/0./
C   data imagen/.false./

```



```

        iycha=objeto(indx+2)+5
C                                     #calcula linea de trayectoria
        call linea(xman,yman,ixcha,iycha,pend,ord)
        call d32(xman,yman,zman,i1,j1)
        call d32(ixcha,iycha,zman,i2,j2)
        call chpage(pagina)
        call workon(pagina)
        call xline(i1,j1,i2,j2,14)
        call chpage(pagina)
        call workon(pagina)
300    continue
        call inkey(tecla)
        if(tecla .eq.esc)go to 10000
C                                     #calcula proxima trayectoria.
        x=xman-impaso
        y=ifix(pend*float(xman)+ord)
C                                     #Vera si hay obstaculo
        call checal(objeto,nobj,x,y,hayobs,indobs)
C                                     #hayobs-HAY OBSTaculo
C                                     #indobs-INDice OBSTaculo.
        go to(400,450),hayobs
C-----
C 121-No hay obstaculo
C-----
400    continue
        call movman(pagina)
        yman=ifix(pend*float(xman)+ord)
        go to 1000
C-----
C 122-Si hay obstaculo
C-----
450    continue
C                                     #Si existe obstaculo,ve de que tipo
C         if(zman.ne.zmin)go to 600
C-----
C 1221-Barrera
C-----
500    continue
        call barrer(pagina,indobs,pend,ord)
C                                     #Ve si la barrera es el refugio
        if(indobs .eq.indcha)go to 2000
C-----
C 1222-Zanja
C-----
600    continue
        call zanja(pagina,indobs,indcha,pend,ord)
        go to 1000
1000   continue
        call escena(pagina)
        go to 300
C-----
C Fin de simulacion
C-----
2000   continue
        call chpage(pagina)

```



```

call workon(pagina)
call logo(x2logo,y2logo,'FIN DE SIMULACIONE')
10000 continue
write(3,10002)
10002 format(// '**** FINAL ****'//)
end

```

```

C
C
C*****
C**  subroutine pinta espacio tridimensional  **
C*****

```

```

C
C      subroutine esptri
C      common/maxim/ixmax,iymax,izmax
C      common/minim/ixmin,iymin,izmin
C      common/comun1/i1,i2,i3,i4
C      common/comun2/j1,j2,j3,j4
C      icolor=15

```

```

C
C Plano X-Y

```

```

C
C      call d32(ixmin,iymin,izmin,i1,j1)
C      call d32(ixmin,iymax,izmin,i2,j2)
C      call d32(ixmax,iymax,izmin,i3,j3)
C      call d32(ixmax,iymin,izmin,i4,j4)
C      call xline(i1,j1,i2,j2,icolor)
C      call xline(i2,j2,i3,j3,icolor)
C      call xline(i3,j3,i4,j4,icolor)
C      call xline(i4,j4,i1,j1,icolor)

```

```

C Plano Y-Z

```

```

C
C      call d32(ixmin,iymin,izmin,i1,j1)
C      call d32(ixmin,iymin,izmax,i2,j2)
C      call d32(ixmin,iymax,izmax,i3,j3)
C      call d32(ixmin,iymax,izmin,i4,j4)
C      call xline(i1,j1,i2,j2,icolor)
C      call xline(i2,j2,i3,j3,icolor)
C      call xline(i3,j3,i4,j4,icolor)
C      call xline(i4,j4,i1,j1,icolor)

```

```

C Plano X-Z

```

```

C
C      call d32(ixmin,iymax,izmin,i1,j1)
C      call d32(ixmin,iymax,izmax,i2,j2)
C      call d32(ixmax,iymax,izmax,i3,j3)
C      call d32(ixmax,iymax,izmin,i4,j4)
C      call xline(i1,j1,i2,j2,icolor)
C      call xline(i2,j2,i3,j3,icolor)
C      call xline(i3,j3,i4,j4,icolor)
C      call xline(i4,j4,i1,j1,icolor)
C      return
C      end

```

```

C
C
C*****
C**  subroutine haz funcion de balanceo  **

```



```

C
subroutine barrer(pagina, indobs, pend, ord)
integer pagina, indobs
real pend, ord
integer objeto(40)
integer xman, yman, zman
integer xmin, ymin, zmin
integer x1logo, y1logo, x2logo, y2logo
logical imagen
common/minim/xmin, ymin, zmin
common/valman/scal, largo, impaso
common/infman/xman, yman, zman, theta, imagen
common/infobj/objeto, nobj
common/valor/pi
common/comun1/i, j, k, l
common/comun2/m, n, iz, ix
common/patman/itip1, idip1, iddp1, itdp1, itip2, idip2, iddp2, itdp2
common/logot/x1logo, y1logo, x2logo, y2logo
data iobclr/7/

C
indx=(indobs-1)*5

C
xman=objeto(indx+1) #asigna nuevos valores a la mantis
yman=ifix(pend*float(xman)+ord)
if(zman.eq.zmin)zman=largo
if(zman.ne.zmin .and. zman.lt.largo)zman=largo
theta=3.*pi/2.
ialtur=objeto(indx+5)

C
if(zman.gt.ialtur+impaso)go to 150 #si el obj. es pequeno no trepa
-----
C Trepa la pared de la barrera
-----
C
100 continue
call escena(pagina)
call movman(pagina)
iz=zman+impaso
if(iz.lt.ialtur)go to 100
-----
C Camina sobre la cara superior de la barrera
C siguiendo la trayectoria al refugio
-----
C
150 continue
theta=0.
ixmin=objeto(indx+1)
ixmax=objeto(indx+3)
iymin=objeto(indx+2)
iymax=objeto(indx+4)
zman=ialtur

C #asigna primero la x superior menos el largo
C #en caso de que esta salga del objeto asigna
C #la x inferior (limite del objeto)
xman=ixmax-largo
if(xman.lt.ixmin)xman=ixmin

```

```

yman=pend*float(xman)+ord
ipoint=(indobs-1)*5+1
call checal(objeto(ipoint),1,xman,yman,iesta,icode)
go to(175,200),iesta
C
C #No cae la primer posicion dentro de la cara
C #Ajustara para que este en el limite de ella
175 continue
    if(yman.gt.iymax)yman=iymax
    if(yman.lt.iymin)yman=iymin
    go to 300
200 continue
    ix=xman-impeso
    iy=pend*float(ix)+ord
    ipoint=(indobs-1)*5+1
    call checal(Objeto(indobs),1,ix,iy,iesta,icode)
    go to(275,250),iesta
C
C #puede caminar sobre la cara todavia
250 continue
    call escena(pagina)
    call movman(pagina)
    yman=pend*float(xman)+ord
    go to 200
C
C #Ya no puede caminar en la cara superior
C #se quedara en los bordes proximos a la tray.
275 continue
    if(ix.gt.ixmin)xman=ix
    if(ix.le.ixmin)xman=ixmin
    yman=pend*float(xman)+ord
    go to 175
300 continue
    j=xman+largo
    if(j.le.ixmax)go to 350
C
C #el espacio para colocar las
C #patas trasera es pequeno
    call defclr(itdp1,0,0,0)
    call defclr(itip1,0,0,0)
    call defclr(itdp2,0,15,0)
    call defclr(itip2,0,15,0)
350 continue
    call escena(pagina)
    continue
    return
    end
C
C
C*****
C**  subroutine haz movimiento con ZANJA      **
C*****
C
subroutine zanja(pagina,indzan,indcha,pend,ord)
integer pagina,indzan,indcha
real pend,ord
integer objeto(40)
integer xman,yman,zman
logical imagen

```

```

integer xmin,ymin,zmin
integer xpaso
common/minim/xmin,ymin,zmin
common/valman/scal,largo,impeso
common/infman/xman,yman,zman,theta,imagen
common/infobj/objeto,nobj
common/valor/pi
common/comun1/i,j,k,l
common/comun2/m,n,iz,ix
common/patman/itip1,idipl,iddp1,itdp1,itip2,idi2,iddp2,itdp2

```

```

C
C CONSTANTES DE SALTO BALISTICO
C Phi=[degradados] ; vz=cm/seg ; g=[cm/seg**2]

data phi/45./,vz/275.00/,g/981./

C
ialmin=2.*float(largo)
ialtur=zman
C Vera cual es el primer objeto que se encuentra si sigue su trayectoria
C al nivel del piso
indx=(indcha-1)*5
ixchl=objeto(indx+3)-5
ixpaso=2
do 20 ii=ixchl,xman,ixpaso
iix=xman-ii+ixchl
iiy=pend*float(iix)+ord+1
call checal(objeto,nobj,iix,iiy,icode,iprim)
if(icode.eq.1)go to 20
ii=xman
20 continue
C Al llegar aqui el primer objeto que se encuentra en su
C trayectoria es el iprim
C
C Aplica la decision de teoria de esquemas
if(ialtur.gt.ialmin)go to 200
C-----
C Baja zanja
C-----
100 continue
C
C # vera si al bajar existe un hueco de tal
C # forma que quepa horizontalmente

ixprimxman=largo-impeso
do 102 ii=ixprim,xman
iix=xman-ii+ixprim
iiy=pend*float(iix)+ord
ipoint=(iprim-1)*5+1
call checal(objeto(ipoint),1,iix,iiy,icode,ii)
if(icode.eq.2)ii=xman
102 continue
go to(110,150),icode
C*****
C Le es posible bajar la Zanja
C*****
110 continue

```

```

theta=pi/2.
iiz=zman-largo
C      #si la Zanja es muy pequena no baja por la pared.Salta
if((iiz-impaso.lt.zmin)go to 130
zman=iiz
120  continue
call escena(pagina)
call movman(pagina)
iiz=zman-impaso
if((iiz.ge.zmin)go to 120
C      #ya llego al piso
130  continue
xman=xman-largo
yman=pend*float(xman)+ord
zman=zmin
theta=0.
go to 1000

C
C      #La Altura de la Zanja es minima,podia bajar pero no habia
C      #      espacio en el piso para ella por haber otro objeto.
C
C      #      a)Que el objeto a dirigirse sea de altura menor a la elevacion
C      #      de la zanja
C      #      b)Que el objeto a dirigirse sea de altura mayor o igual que la
C      #      elevacion de la zanja.
C      #      Por tanto las acciones a llevar a cabo son:
C      #      para:
C      #      a) Bajar de ser posible hasta el nivel del otro objeto
C      #      y pasar a el.
C      #      b) Pasar al otro objeto.
150  continue
indx=iprim*5
if(objeto(indx).ge.zman)go to 180
C+++++
C      Caso (a): Baja y pasa al otro objeto
C+++++
theta=pi/2.
iiz=zman-largo
izobj=iprim*5
izobj=objeto(izobj)
C      #si la diferencia entre el nivel de la Zanja y el de el
C      #      otro objeto.no baja por la pared,si no salta.
if((iiz-impaso.lt.izobj)go to 160
zman=iiz
155  continue
call escena(pagina)
call movman(pagina)
iiz=zman-impaso
if((iiz.ge.izobj)go to 155
C      #ya llego a la altura del otro objeto
160  continue
C      #en iix e iiy esta el primer punto de interseccion
C      #de la trayectoria con el objeto
indx=iprim*5
xman=iix

```

```

yman=liy
zman=objeto(indx)
theta=0.
go to 1000

```

```

C+++++
C Caso (b): Pasa al otro objeto
C+++++
180 continue

```

```

C
C #en lix e liy esta el primer punto de interseccion
C #de la trayectoria con el objeto

```

```

xman=lix
yman=liy
theta=0.
go to 1000

```

```

C
C -----
C Ejecucion de Salto Balistico
C -----

```

```

C
C 200 continue
C #Funcion para V0 obtenida de la observacion
C #para obtener un buen parametro de acuerdo
C #a la resolusion, y es aprox. al quintuple
C #de la V0 real de acuerdo al tamaño

```

```

V0=Vz+Vz*scal/2.
ang=phi*pi/180.
a1=sin(ang)/cos(ang)
o2=g/[2.*(V0*cos(ang))**2]
indx=(indcha-1)*5
ixchl=objeto(indx+3)-5
ipega=0

```

```

C
C Calcula el valor de X para cuando este en el suelo que es el maximo
C alcance de su salto.

```

```

liz=-zman
ixmax=(a1+sqrt(a1**2-4.*a2*liy))/(2.*a2)
ixmax=xman-ixmax
if(ixmax.lt.ixchl)ixmax=ixchl

```

```

C
C Siguiendo la trayectoria parabolica, encontrara su punto final, que puede
C ser el piso o un objeto(inclusive el refugio).

```

```

C # nn=el no. de objetos que estan frente a la mantis
C # ni=el 1er. objeto que esta frente a la mantis

```

```

ni=indzan+1
call chpage(pagina)
call workon(pagina)
do 210 ii=ixmax,xman
ix=xman-ii+ixmax
ix=xman-ii
iz=a1*float(ix)-a2*float(ix)**2
iiz=zman+iz
iiy=pend*float(iix)+ord

```

```

call d32(iix,iiy,iiz,ia,ib)
call xdot(ia,ib,1)
do 205 ij=n1,nobj
  ipoint=(ij-1)*5+1
  call checal(objeto(ipoint),1,iix,iiy,icode,it)
  if(icode.eq.1)go to 205

```

C #Hay que ver si se intercepta en la altura
 C #del objeto.

```

  lobsta=ij
  indx=lobsta*5
  if(iiz.gt.objeto(indx))goto 205
  ipega=1
  ij=nobj
  ii=xman

```

```

205 continue
210 continue
call chpage(pagina)
call workon(pagina)

```

C #una vez que se ha localizado el punto final
 C #del salto,se ejecuta dibujando la mantis

```

ifirst=iix
ilast =xman
jj=(ilast-ifirst)/5+1
izalt=zman
theta=340.*pi/180.
call chpage(pagina)
call workon(pagina)
do 220 ii=ifirst,ilast,jj
  iix=ilast-ii+ifirst
  ix=ilast-iix
  iz=a1*float(ix)-a2*float(ix)**2
  iiz=izalt+iz
  iiy=pend*float(iix)+ord
  xman=iix
  yman=iiy
  zman=iiz
  call drwman
220 continue
xman=ifirst
yman=pend*float(xman)+ord
if(ipega.eq.1)go to 225

```

C Cayo en el Piso.

```

  zman=zmin
  theta=0.
  go to 230

```

C Cayo en un objeto. En una cara superior o lateral?

C 225 continue #calcula el punto anterior al de interseccion

```

  iix=xman+1
  iiy=pend*float(iix)+ord
  ipoint=(lobsta-1)*5+1

```



```

call checal(objeto(ipoint),1,lix,liy,icode,ii)
go to(228,227),icode
C                                     #Llego por arriba,esta sobre la cara superior
227  continue
      theta=0.
      indx=iobsta*5
      zman=objeto(indx)
      go to 229
C                                     #Llego por los lados,esta en una cara lateral
228  continue
      theta=3.*pi/2.
      go to 229
229  continue
230  continue
      call chpage(pagina)
      call workon(pagina)
      go to 1000
300  continue
1000 continue
      return
      end

C
C*****
C**  subroutine haz movimiento de Mantis      **
C*****
C
C      subroutine movman(pagina)
      integer ixo,iyo,izo
      real theta
      logical imagen
      integer pagina

C
C      common/valman/scal,largo,impaso
      common/patman/itip1,idip1,iddp1,itdp1,itip2,idip2,iddp2,itdp2

C
C      common/infman/ixo,iyo,izo,theta,imagen
      common/valor/pi

C
C      data iclman/5/,inegro/00/
      data ntime/30000/

C
C El movimiento de las patas se hara en la escena ya desplegada
C
C      call chpage(pagina)
      call workon(pagina)

C
C La forma de dar movimiento es para cada una de las patas en movimiento:
C a) Borrar la pata en posicion anterior (estatica o posicion 1)
C b) Pintar la pata en nueva posicion (dinamica o posicion 2)
C
C      nmov=0
10   continue
      nmov=nmov+1
      go to(100,200,300,400,500),nmov
C-----

```

C Posicion 1 : pata delantera derecha

```

C-----
C
100  continue
      call delay(ntime)
      call defclr(iddp1,0,0,0)
      call defclr(iddp2,0,15,0)
      go to 10
    
```

C Posicion 2 : pata trasera Izquierda

```

C-----
C
200  continue
      call delay(ntime)
      call defclr(itip1,0,0,0)
      call defclr(itip2,0,15,0)
      go to 10
    
```

C Posicion 3 : pata delantera Izquierda

```

C-----
C
300  continue
      call delay(ntime)
      call defclr(idip1,0,0,0)
      call defclr(idip2,0,15,0)
      go to 10
    
```

C Posicion 4 : Modificación de coordenadas
para nueva posición de Mantis
en posición de reposo

```

C-----
C
400  continue
      if(theta.eq.pi/2.)go to 420
      if(theta.eq.3.*pi/2.)go to 430
    
```

C decremento en X

```

C
410  continue
      ixo=ixo-impaso
      go to 10
    
```

C decremento en Z

```

C
420  continue
      izo=izo-impaso
      go to 10
    
```

C incremento en Z

```

C
430  continue
      izo=izo+impaso
      go to 10
    
```

C regresa a la pagina que se tenia originalmente

C

```

500 continue
    call chpage(pagina)
    call workon(pagina)
    return
end

```

```

*****
**      subroutine ordena Objetos definidos      **
*****

```

Ordena Los objetos definidos de mayor a menor en las X maybres como primer llave y de mayor a menor en las Y mayores como segunda llave para de esta forma poder manipular los espacios y movimientos de la Mantis Religiosa.

```

subroutine ordena(objeto.nobj)
integer objeto(1)
if(nobj.lt.2)go to 500

```

Ordena por primer llave (X's superiores de cada objeto)

```

call sort(objeto.nobj,3)

```

Ordena por segunda llave(Y's superiores de cada objeto)

```

do 50 i=1.nobj
  ii=(i-1)*5+3
  do 20 j=i.nobj
    jj=(j-1)*5+3
    if(objeto(jj).ne.objeto(ii))go to 30
20 continue
  j=j+1
30 continue
  j=j-1
  n=j-i+1
  call sort(objeto(i).n,4)
  i=j
50 continue
500 continue
return
end

```

```

*****
**      subroutine aplica Sort Insercion Directa      **
*****

```

Ordena Los objetos definidos de mayor a menor tomando como llave ipos de cada objeto

```

ipos= 1 xmenor
      = 2 ymenor
      = 3 xmayor
      = 4 ymayor
      = 5 altura

```

C
 C METODO DE INSERCIÓN DIRECTA.
 C

```

C
subroutine sort(objeto,nobj,ipos)
integer objeto(1)
integer aux(5)
common/comun1/ielem,iant,iscan,ixscan
common/comun2/j,k,l,m
if(nobj.lt.2)go to 500
do 40 ielem=2,nobj
    iant=ielem-1
    ixelem=(ielem-1)*5+ipos
    do 20 j=1,iant
        iscan=iant-j+1
        ixscan=(iscan-1)*5+ipos
        if(objeto[ixelem].lt.objeto[ixscan])go to 25
20    continue
    iscan=0
25    iscan=iscan+1
    if(iscan.eq.ielem)go to 40
    k=(ielem-1)*5+1
    l=ielem*5
    i=0
    do 28 j=k,1
        i=i+1
        aux(i)=objeto(j)
28    continue
    do 30 j=iscan,iant
        i=iant-j+iscan
        k=(i-1)*5+1
        l=i*5
        do 29 m=k,1
            objeto[m+5]=objeto(m)
29    continue
30    continue
    k=(iscan-1)*5+1
    l=iscan*5
    i=0
    do 35 j=k,1
        i=i+1
        objeto(j)=aux(i)
35    continue
40    continue
500    continue
return
end
    
```

C
 C
 C*****
 C** subroutine cambia el indicador de pagina **
 C*****
 C

```

subroutine chpage(pagina)
integer pagina
    
```

```

pagina=pagina+1
pagina=mod(pagina,2)
return
end

```

```

C
C
C*****
C**  subroutine pinta escena de simulacion  **
C*****
C
  subroutine escena(pagina)
  integer pagina
  integer objeto(40)
  integer xman,yman,zman
  logical imagen
  common/valman/scal,largo,impaso
  common/infman/xman,yman,zman,theta,imagen
  common/infobj/objeto,nobj
  data iobclr/7/

C
  call esptri
  call drwobj(objeto,nobj,iobclr)
  call drwman
  call animat
  call chpage(pagina)
  return
  end

```

A P E N D I C E 8

P R O G R A M A A N A L I S I S S A L T O B A L I S T I C O .


```
data ba,bz/'a','z'/
data icolor/4/
```

```
#
#
call grafix
pi=atan2(0,-1)
raddeg=180./pi
repeat
(
```

```
#-----
# interaccion
#-----
```

```
10 write(icrt,10)
format('//'+ (1)leer de archivo (2)leer de terminal ')
if(haydat)
write(icrt,20)
20 format('//'+ (3)graficar datos (4)ajustar datos ')
if(ajust)
if(hayaj)
write(icrt,30)
else
write(icrt,40)
30 format('//'+ (5)aplica ec. Ajuste (6)graficar datos ajust. ')
40 format('//'+ (5)aplica ec. ajuste ')
write(icrt,50)
50 format('//'+ ..... ')
read(icrt,115)icmd
go to(100,200,300,400,500,300), icmd
go to 10000
```

```
#-----
# LEER ARCHIVO
#-----
```

```
100 continue
write(icrt,105)
105 format('+',10x,'Nombre de archivo ..... ')
read(icrt,109)(file(1),i=1,8)
108 format(8a1)
for (i=1 ; i <= 8 ; i=i+1)
(
if(file(i).ge.ba .and. file(i).le.bz)
file(i)=file(i)-32
)
call open(ifile,file,0)
input=ifile
110 read(input,115,end=190)j
115 format(15)
do i=1,j
(
if(input .eq. icrt)
write(icrt,118)
118 format('+?')
read(input,120,end=195)x(i),y(i)
119 format(2e20.2)
120 format(2e10.2)
```



```

        if(input .eq. ifile)
            write(icrt,520)1,x(i),1,y(i)
        }
        haydat=.true.
        ajust=.false.
        hayaj=.false.
        n=j
        write(icrt,125)N
125      format(/10x,' Numero de datos : ',14/)
        endfile input
        go to 10000
#
190      write(icrt,191)
191      format(' --- No Existe Archivo ---'/)
        endfile ifile
        go to 10000
#
195      n=i-1
        write(icrt,196)N
196      format(' --- Faltan datos. N = ',14,' ---'/)
        haydat=.true.
        ajust=.false.
        hayaj=.false.
        endfile ifile
        go to 10000
#
#-----
# LEER DATOS POR TERMINAL
#-----
200      continue
        write(icrt,201)
201      format('+',10x,'Numero de Parejas .... ')
        input=icrt
        go to 110
#
#-----
# GRAFICA DATOS
#-----
300      continue
        write(icrt,302)
302      format('+',10x,'(0)Limpia Pantalla (1)Misma Pantalla ....')
        read(icrt,115)i
        icall = i+1
        if(icall .eq. 1)
            (
                write(icrt,305)xmin,ymin
305      format('+',10x,'XMIN = ',e10.2,' YMIN = ',e10.2,' ..... ')
                read(icrt,120)x1,y1
                if(x1.ne.0.)
                    xmin=x1
                if(y1.ne.0.)
                    ymin=y1
                write(icrt,310)xmax,ymax
310      format('+',10x,'XMAX = ',e10.2,' YMAX = ',e10.2,' ..... ')
                read(icrt,120)x1,y1
            )
        )
    
```

```

if(x1.ne.0.)
  xmax=x1
if(y1.ne.0.)
  ymax=y1
write(icrt,315)xinc,yinc
315 format('+',10x,'XINC= ',e10.2,' YINC = ',e10.2,' ..... ')
read(icrt,120)x1,y1
if(x1.ne.0.)
  xinc=x1
if(y1.ne.0.)
  yinc=y1
)
write(icrt,325)
325 format('+',10x,'{0}Puntos {1}Lineas ..... ')
read(icrt,115)i
iline=i

*
if(icmd.eq.6)
  go to 500

*
call curvas(x,y,n,icall,xmin,xmax,xinc,ymin,ymax,yinc,icolor,iline)
go to 10000

#-----
# AJUSTA DATOS
#-----
400 continue
   igrad=2
   call regres(x,y,n,coef,igrad)
   call stat(x,y,n,sumx,sumy,sumx2,sumy2,sumxy,sumx2y)

*
# obtiene error estandard y coeficiente de correlacion
*
syx$x2=sumy2-(coef(1)*sumy+coef(2)*sumxy+coef(3)*sumx2y)
syx$y2=sqrt(abs(syx$x2)/n)
ryx$x2=coef(1)*sumy+coef(2)*sumxy+coef(3)*sumx2y-sumy*sumy/n
ryx$y2=ryx$x2/(sumy2-sumy*sumy/n)
ryx$xy=sqrt(abs(ryx$y2))

*
write(icrt,415)igrad
415 format(/20x,' Coeficientes y estadistica del ajuste de Grado ',i3,
/20x,' -----')
j=igrad+1
for (i=1 ; i <= j ; i=i+1)
  (
    l=i-1
    write(icrt,420)l,coef(l)
420 format(20x,' A[',i2,']=',f10.4)
  )
  write(icrt,425)syx$y2,ryx$xy
425 format(30x,'Sx.y,x2 = ',f10.4,' rx.y,x2 = ',f10.4)

*
# Obtiene la parabola de 2 terminos (el eliminar el t. independiente
# implica que pasa por el origen)
#
k=n-1

```

```

do i=1,k
if(x[i].ne. 0. .and. x[n].ne. 0.)
if(x[i].ne. x[n])
{
y1=coef[1]+coef[2]*x[i]+coef[3]*x[i]*x[i]
y2=coef[1]+coef[2]*x[n]+coef[3]*x[n]*x[n]
a1=y1/x[i]-y2*x[i]/(x[n]*x[n])
a1=a1/(1-x[i]/x[n])
a2=(y1-a1*x[i])/(x[i]*x[i])
i=k
}
#
# Obtene los parametros del salto balistico de acuerdo a la recta.
#
teta =atan(a1)
t=cos(teta)
t=t*t
t=981./(2*t*a2)
Vo=sqrt(abs(t))
teta =raddeg*teta
write(icrt,430)a1,a2,teta,vo
430 format//20x,' Coeficientes de Parabola sin termino independiente',
/20x,'
/20x,' a1 = ',f10.4,', a2 = ',f10.4//,
/20x,' Valores de salto balistico',
/20x,'
/20x,' Teta = ',f10.4,', Vo = ',f10.4)
ajust=.true.
go to 10000
#-----
# GENERA DATOS DE ECUACION DE AJUSTE
#-----
500 continue
write(icrt,505)
505 format('+',10x,'X inicial ... ')
read(icrt,120)x1
write(icrt,510)
510 format('+',10x,'X final ... ')
read(icrt,120)xn
write(icrt,515)
515 format('+',10x,'Numero de puntos a generar ..... ')
read(icrt,115)najust
if(najust.gt.100)
najust=100
xi=(xn-x1)/(najust-1)
do i=1,najust
{
xaj[i]=x1
yaj[i]=a1*xaj[i] + a2* xaj[i]* xaj[i]
#
520 write(icrt,520)i,xaj[i],1,yaj[i]
format(' ',10x,'X(',i3,')= ',f10.3,', Y(',i3,')= ',f10.3)
x1=x1+xi
}
hayaj=.true.
go to 10000

```

```

#-----
# GRAFICA DATOS AJUSTADOS
#-----
600   continue
      i=icolor+10
      call curvas(xaj,yaj,najust,icall,xmin,xmax,xinc,ymin,ymax,yinc,
                 i,iline)
      go to 10000

#
#
10000 continue
      }
      until (icmd .eq.0)
      end

#
#
*****
*** SUBROUTINE ESTADISTICAS 2 VECTORES      **
***                                         **
*****
#
#
subroutine stat(x,y,n,sumx,sumy,sumx2,sumy2,sumxy,sumx2y)
real x(1),y(1)
sumx=0.
sumy=0.
sumx2=0.
sumy2=0.
sumxy=0.
sumx2y=0.
do i=1,n
(
  sumx=sumx+x(i)
  sumy=sumy+y(i)
  sumx2=sumx2+x(i)*x(i)
  sumy2=sumy2+y(i)*y(i)
  sumxy=sumxy+x(i)*y(i)
  sumx2y=sumx2y+x(i)*x(i)*y(i)
)
return
end

```

```

C ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C :::::::::::::::::::::
C ::::::::::::::BIBLIOTECA REGRESION lineal y no-lineal::::::::::::::::::
C :::::::::::::::::::::
C ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C
C      (c)Vicente Alcantara G.
C      09-ago-83
C
C      LENGUAJE: Fortran
C
C
C*****
C** SUBROUTINE OBTEN PARAMETROS DE REGRESION DE GRADO n      **
C**                                                         **
C*****
C
      subroutine regres(x,y,n,coef,igrad)
      real x(1),y(1),coef(1)
      real a(10,10),b(10,10),c(10,10)
C
C      Subrutina que realiza el ajuste a una curva por el metodo de minimos
C      cuadrados.
C
C      PARAMETROS:
C
C      X          Vector que contiene los datos de la var. independiente.
C      Y          Vector que contiene los datos de la var. dependiente.
C      N          Numero de parejas a tomar en el ajuste.
C      COEF       Vector que tendra los coeficientes de la curva de ajuste.
C      IGRAD      Grado de la curva de ajuste.
C
C      Nota: Se regresaran IGRAD+1 valores en COEF
C
      ndim=igrad+1
      call ecnorm(x,y,n,a,b,igrad)
      call invmat(a,c,ndim,icode)
      if(icode .ne. 0)go to 1000
      call multx(c,b,a,ndim,ndim,1)
      do 10 i=1,ndim
         coef(i)=a(i,1)
      10 continue
      1000 return
      end
C*****
C** SUBROUTINE FORMA ECUACIONES NOMALES A PARTIR DE CRITERIO **
C** DE MINIMOS CUADRADOS **
C*****
C
      subroutine ecnorm(x,y,n,a,b,igrad)
      real x(1),y(1)
      real a(10,10)
      real b(10,10)
C

```

C Subrutina que forma las ecuaciones normales en base a N datos y a el
 C orden de ajuste.

C
 C PARAMETROS :
 C
 C X Vector con los datos de la Var. independiente.
 C Y Vector con los datos de la Var. dependiente.
 C N Numero de parejas (X,Y) a tomar.
 C A Matriz de dimension [IGRAD+1 , IGRAD+1] donde se ponen
 C terminos dependientes de las ecs. normales
 C B Matriz de dimension [IGRAD+1 , 1] donde se ponen los
 C terminos independientes de las ecs. normales.
 C IGRAD Grado de las ecs. Normales.

```

C      ndim=igrad+1
C      m=ndim+2
C      do 500 j=1,m
C          b(j,1)=0.
C          do 600 i=1,n
C              b(j,1)=b(j,1)+x(i)**(j-1)
500      continue
500      continue

C      do 700 i=1,ndim
C          pondra los terminos de las columnas
C          k=i
C          do 900 j=1,n
C              a(i,j)=b(k,1)
C              k=k+1
900      continue
C          pondra los terminos independientes de la matriz aumentada
C          b(i,1)=0.
C          do 800 j=1,n
C              b(i,1)=b(i,1) + y(j) * x(j) ** (i-1)
800      continue
700      continue

C      return
C      end
    
```

C
 C*****
 C** SUBROUTINE INVIERTE MATRIZ **
 C**
 C*****

```

C      subroutine invmat(c,b,n,icode)
C      real c(10,10),b(10,10)
C      real a(10,10)
C      data ndim/10/

C      Subrutina que obtiene la inversa de una matriz.
C      C
C      PARAMETOS :
C      C
C      C          matriz a obtener su inversa
    
```

```

C      B      matriz inversa de C
C      N      dimension de C
C      ICODE  resultado de la llamada de esta rutina.
C              =0      O.K.
C              =1      error en la diagonalizacion; proceso incompleto
C
C      icode =1
C      if(n.le ndim)go to 50
C      write(3,10)
10    format(' ** D/INV **')
C      go to 1000
C
C forma matriz identidad y [A]=[C]
50    continue
C      do 100 i=1,n
C          do 150 j=1,n
C              a(i,j)=a(i,j)
C              b(i,j)=0
150    continue
100   continue
C      do 200 i=1,n
C          b(i,i)=1
200   continue
C
C empieza el metodo de eliminacion de Gauss
C      do 500 ii=1,n
C          piv=a(ii,ii)
C          if(piv.eq. 0.)go to 1000.
C          do 250 j=1,n
C              a(ii,j)=a(ii,j)/piv
C              b(ii,j)=b(ii,j)/piv
250   continue
C          if(ii.eq.n)go to 500
C          k=ii-1
C          do 300 i=k,n
C              fact=a(i,ii)
C              do 350 j=1,n
C                  a(i,j)=a(i,j)-a(ii,j)*fact
C                  b(i,j)=b(i,j)-b(ii,j)*fact
350   continue
300   continue
500   continue
C
C Hasta aqui A es una matriz triangular superior; se procedera a hacerla
C identidad, lo cual implica que B sea la inversa
C
C      m=n-1
C      do 800 ii=1,m
C          ir=m-ii+1
C          kr=ir+1
C          do 700 ic=k,n
C              ireng=n-ic+k
C              fact=a(ir,ireng)
C              do 750 j=1,n
C                  a(ir,j)=a(ir,j)-a(ireng,j)*fact

```

```

        b(ir,j)=b(ir,j)-b(ireng,j)*fact
750    continue
700    continue
800    continue
C
    icode = 0
1000  continue
    return
    end
C*****
C**          SUBROUTINE MULTIPLICA DOS MATRICES          **
C**                                                              **
C*****
C
    subroutine multX(a,b,c,i,j,k)
    real a(10,10),b(10,10),c(10,10)
C
C  Subrutina que realiza la Multiplicacion de 2 matrices.
C
C  PARAMETROS :
C
C  A          1er. operando de operacion matricial
C  B          2o. operando de operacion matricial
C  C          Resultado de operacion Matricial
C  I          No. de renglones de A
C  J          No. de columnas de A y de Renglones de B.
C  K          No. de columnas de B.
C
    integer i,j,k
    do 30 ii=1,i
    do 20 kk=1,k
    c(ii,kk)=0.
    do 10 jj=1,j
    c(ii,kk)=c(ii,kk)+a(ii,jj)*b(jj,kk)
10    continue
20    continue
30    continue
    return
    end

```



```

C          valor minimo corresp. de los datos a graficar.
C  XN,YN          Cotas mayores a graficar *
C                si alguno es mayor o igual a 1.e15 se toma el
C                valor maximo corresp. de los datos a graficar
C  XI,YI          Incremento entre cada espacio de la pantalla de
C                los puntos a graficar.
C  ICOLOR         Codigo de color a graficar la curva
C  ILINE         Indica si se hace la grafica con puntos o
C                con lineas.
C                =0 grafica con puntos
C                =1 grafica con lineas
C  * Estos parametros solo son tomados en cuenta cuando ICALL=1 ,es
C    decir en la primer llamada
C
C  integer pagsiz(2).offset(2)
C  byte number(13)
C  common/perm/xmin,xmax,xinc,xming,xmaxg,ixof,ixdesp,
C  *          ymin,ymax,yinc,yming,ymaxg,iyof,iydesp
C
C  data pagsiz/330,220/,offset/378,241/
C  data icotx/5/,icoty/5/
C  data tope/1.e15/
C  data number/'0','0'/'
C
C  if(icall ne 1)go to 100
C es la primer llamada.por tanto se escalara el espacio de graficacion y se
C pintaran los ejes.asi como sus acotaciones
C
C  ncatx=pagsiz(1)
C  ncaty=pagsiz(2)
C  ldeso=offset(1)-ncatx
C  iydesp=offset(2)-ncaty
C tomara inicialmente valores minimos y maximos de los arreglos X e Y
C  call sigma(x,n,xmin,xmax,i,j)
C  call sigma(y,n,ymin,ymax,i,j)
C  if(x1.gt.-tope)xmin=x1
C  if(xn.lt.tope) xmax=xn
C  if(y1.gt.-tope)ymin=y1
C  if(yn.lt.tope) ymax=yn
C  xinc=x1
C  yinc=y1
C  if(xi .eq.0.)xinc=(xmax-xmin)/(ncatx-1)
C  if(yi .eq.0.)yinc=(ymax-ymin)/(ncaty-1)
C XMING,YMING,XMAXG,YMAXG son los valores minimo y maximo graficables en X,Y
C
C  10  nran=abs((ymax-ymin)/yinc)+1
C      if(nran.gt.ncaty)nran=ncaty
C      ymaxg=yinc*(nran-1)+ymin
C      yming=ymin
C      iyof=0
C      if(ymax .gt.ymin)go to 15
C se graficara en forma inversa el eje Y
C      if(yinc .gt.0.)yinc=-yinc
C      ymaxg=ymin

```

```

yming=yinc*(nran-1)+ymin
iyof=nran
iyof=iyof+1
15 continue
nran=abs((xmax-xmin)/xinc)+1
if(nran .gt. ncatx)nran=ncatx
xmaxg=xinc*(nran-1)+xmin
xming=xmin
ixof=0
if(xmax.gt.xmin)go to 20
C se graficara en forma inversa el eje X
if(xinc .gt. 0.)xinc=-xinc
xmaxg=xmin
xming=xinc*(nran-1)+xmin
ixof=nran
ixof=ixof+1
20 continue
C se inicializara la pantalla
call init
call init1
C se graficaran los ejes
call xline(ixdesp,lydesp,ncatx+ixdesp,lydesp,15)
call xline(ixdesp,lydesp,ixdesp,ncaty+iydesp,15)
C se acotaran los ejes
C ---X---
i=ncatx/icotx
do 25 j=1,ncatx,i
k=j+ixdesp-1
call xline(k,lydesp,k,lydesp-3,15)
25 continue
C ---Y---
i=ncaty/icoty
do 30 j=1,ncaty,i
k=j+iydesp-1
call xline(ixdesp,k,ixdesp-3,k,15)
30 continue
C
C se pondran las acotaciones en los ejes
call resbox(1,1,24,lydesp)
j=ixdesp/15+1
call resbox(1,1,j,34)
call hscale(2.755 1,48)
C ---X---
j=ncatx/icotx
do 40 i=1,icotx
xx=(i-1)*j*xinc+xming
encode(number,45)xx
k=(i-1)*ncatx/icotx+ixdesp/2
k=k*2
call hxttext(k,lydesp-9,15,number)
40 continue
45 format(g9.3,'0')
C ---Y---
j=ncaty/icoty
do 50 i=1,icoty

```



```
C
      integer function iposic(x,xmin,xinc,xming,xmaxg,ixof,ixdesp)
C
C  funcion que da la posicion de una de las coordenadas de graficacion
C  en funcion de la escala
C
```

```
      real x,xmin,xinc,xming,xmaxg
      integer ixof
      ixpos=0
      if(x .lt. xming .or. x.gt. xmaxg)go to 100
      ixpos=abs((x-xmin)/xinc)+1
      ixpos=iabs(ixof-ixpos)
      ixpos=ixpos+ixdesp
100  iposic=ixpos
      return
      end
```

```
C
C*****
C** SUBROUTINE  OBTEN SUMATORIA          **
C**                                          **
C*****
C
```

```
.subroutine sigma(x,n,xmin,xmax,ixmin,ixmax)
      real x(1)
      xmin=x(1)
      xmax=x(1)
      ixmin=1
      ixmax=1
      do 10 i=2,n
          if(x(i).gt.xmin)go to 15
          xmin=x(i)
          ixmin=i
15      continue
          if(x(i).lt.xmax)go to 10
          xmax=x(i)
          ixmax=i
10      continue
      return
      end
```

APENDICE C.
PROGRAMA DE SIMULACION DE ATAQUE.

```

*****
*****
*****          P R O G R A M A      S I M A T A Q          *****
*****
*****
#
# Simulacion del modelo del ataque de la Mantis Religiosa ante una presa.
#
#
# (C) Vicente Alcantara Guerrero.
#
# feb/10/1984
#
# Lenguaje: Ratfor de Cromemco.
#
# IMPORTANTE: Es necesario tener en la configuracion del
#             sistema una interface de graficacion SDI con
#             2 paginas de 48Kb y un monitor de color RGB.
#
# Bibliotecas ligadas:
#
# Fgraf1
# Graf2
# definem
# manlib1
# crtlib
# forlib
#
#
# program ataque
# integer x,y,deltax,deltay,ataca
# integer xz,yz,dist
# real newang
# common /random/sem
# common /esp/pi
# common /salida/irep,iout
# data irep/0/ , iout/2/ , icrt/3/
# data anglim/135./
# data longp/120/
#
#
# call grafix
# pi=atan2(0,-1)
# raddeg=180./pi
#
# obtencion de la "semilla" para la rutina de numeros aleatorios.
#
# call time(i,j,k)
# si=i+1
# sj=j+1
# sk=k+1
# sem=(si/60.)+(sj/60.)*(1./1.E-3))+((sk/24.)*(1./1.E-4))
#
# valores para resolucion media.
#
#
# X#max=378
# Y#max=241

```



```

    }
    write(iout,20)longp
20    format(' ---- SIMULACION ATAQUE A PRESA POR LA M.R. ---- ',
           //,' MANTIS RELIGIOSA.',
           //,' Posicion : X=0 Y=0',
           //,' Long. Patas: ',i5//)
    }

#
# obtiene coordenadas iniciales para la posicion de la presa (X e Y)
#
    X=ifix[Rnd(sem)*Limx]-ifix[Rnd(sem)*Limx]
    Y=ifix[Rnd(sem)*Limy]-ifix[Rnd(sem)*Limy]
#
#-----
# SIMULACION DEL MODELO
#-----
#
    Repeat
    {
#           define los colores de graficacion
        call defclr(14,12,14,0)
        call defclr(12,15,8,0)
        call defclr( 5,0,15,0)
        call defclr( 9,15,0,0)
#           hace el movimiento aleatorio del punto
        Call Coord(X,Y,Deltax,Deltay,limx,limy)
        x1=float(x)
        y1=float(y)
        dist=sqrt(x1*x1+y1*y1)
#           calcula el angulo de la presa
        angpre=atan2(y1,x1)
#           dibuja la presa
        call xfcir(X,Y,2,15)
        r1=15.
        if(r1 .gt.dist)r1=dist/2.
        iy1pr=r1*sin(angpre)
        ix1pr=r1*cos(angpre)
#           trabaja el angulo de la presa en grados
        angpre=angpre*raddeg
#
# conversion del angulo a impulso nervioso
#
# Escala: 135 grados = 1.0 unidades
# -135 grados = -1.0 unidades
# 0 grados = 0 unidades
#
        if(abs(angpre) <= anglim)
        {
            if(irep == 1)
            {
                write(iout,110)x,y,angpre
                format(//,' Presa: X= ',i4,' Y= ',i4,' ANG. = ',f6.1,
                       //5x,' ANG. CAB.', ' ANG. PAT.', ' SUMA',
                       //5x,'-----','----->')
            }
        }
    }

```

110

```

    call sigue(angpre,angcab,angpat)
    angdib = angcab/raddeg
#           dibuja linea agulo de presa
    call xline(ixlpr,iylpr,x,y,14)
#           dibuja linea angulo de cabeza
    iy1cab=r1*sin(angdib)
    ix1cab=r1*cos(angdib)
    r=dist
    repeat
    {
        iy2cab=r*sin(angdib)
        ix2cab=r*cos(angdib)
        r=r-1
    }
    until (iabs(iy2cab)<=limy & iabs(ix2cab)<=limx)
    call xline(ix1cab,iy1cab,ix2cab,iy2cab,12)
}
Call Figura(Xz,Yz,Factor,angdib,Ataca,X,Y)
call animat
}
Until (Dist<=Longp & abs(angpre) <= anglim)
Ataca=1
call xfcir(x,y,2,15)
Call Figura(Xz,Yz,Factor,angdib,Ataca,X,Y)
call animat
write(iout,120)dist
120 format(//' *** CAPTURA DE PRESA ***   Distancia = ',15,
//' ---- FIN SIMULACION ----')
endfile iout
End
#
#*****
#           SUBROUTINE SIGUE
#*****
#
#           SUBROUTINE SIGUE(Z,Y,X)
#
#
#
# ENTRADA:
#           Z posicion de la presa(grados).
#           Y posicion de la cabeza(grados).
#
# SALIDA :
#           Y nueva posicion de la cabeza(grados)
#           X posicion de ataque de las patas delanteras(grados)
#
#
# Rutina que obtiene el angulo de la cabeza y el de
# las patas delanteras en el seguimiento de una presa,
# considerando el error visual y la contribucion de
# los mecanoreceptores.
# Todos los angulos estan referidos al eje del cuerpo de la
# mantis religiosa a exepcion del de ataque de las patas delanteras
# que esta referido a la cabeza.
#
#

```

* El modelo aplicado esta en funcion a la logica seguida en el
 * modelo de control de Mittelstaedt.

* El modelo a aplicar es el siguiente.

```
*
* Error=Z-Y
* dy/dt=(Error - Error Mecanorec.)
* Error Mecanorecep=y*0.15
*
```

```
*
* donde: Z angulo de la presa
*         Y angulo de la cabeza
*
```

```
*
* COMMON/SALIDA/IREP,IOUT
* DATA PROP/0.15/
* DATA DELTAT/0.05/
* DATA TFIN/10./
* TIME=0.
10  CONTINUE
* E=Z-Y
* DY=E-EM
* EM=Y*PROP
* Y=Y+DY*DELTAT
* sum=y+em
* IF (IREP ==1)
* (
20  WRITE(IOUT,20)Y,EM,SUM
*   FORMAT(5X,3F10.4)
* )
* IF(TIME.GE.TFIN)GO TO 100
* TIME=TIME+DELTAT
* GO TO 10
100 CONTINUE
* X=EM
* RETURN
* END
*
```

```
*
* *****
* SUBROUTINE COORDENADAS
* *****
*
```

* ENTRADA:

```
*
*   XA,YA      : Coordenadas del punto a mover.
*   DX,DY      : Incremento en X e Y a considerar en el mov.
*   LIMX,LIMY  : Valor maximo absoluto a considerar en X e Y.
*
```

* SALIDA :

```
*
*   XA,YA      : Nuevas coordenadas del punto.
*
```

```
*
* Rutina que obtiene las coordenadas de un punto que
* tiene movimiento aleatorio.
*
```

```

#
# Esta rutina distribuye los numeros aleatorios en
# una distribucion normal.
#
SUBROUTINE COORD(XA,YA,DX,DY,LIMX,LIMY)
  integer Xa,Ya, Dx, Dy
  common /random/sem
  data Ndir/8/

  Desv=0.5
  Deltas=0 1
  Repeat
  (
    U1=Rnd(sem)
    U2=Rnd(sem)
    V1=2*U1-1
    V2=2*U2-1
    S=V1**2+V2**2
  )
  Until {S<1.}
  R=Sqrt(abs(2*Alog(S)/S))
  Z1=V1*R
  X1=Z1*Deltas+Desv
# obtiene la direccion de movimiento de acuerdo con el numero aleatorio
# obtenido. Tiene 8 posibilidades de movimiento :
#
#      Y ^      8   1   2
#      |
#      |      7   +   3
#      |
#      |      6   5   4
#      |-----> X
#
  N=Ndir*X1
#
# de acuerdo al Valor de N, se tendran los nuevos valores de X e Y.
#
101  go to(101,102,103,104,105,106,107,108),N
     Ya=Ya+Dy
     go to 110
102  Ya=Ya+Dy
     Xa=Xa+Dx
     go to 110
103  Xa=Xa+Dx
     go to 110
104  Xa=Xa+Dx
     Ya=Ya-Dy
     go to 110
105  Ya=Ya-Dy
     go to 110
106  Xa=Xa-Dx
     Ya=Ya-Dy
     go to 110

```



```

data cabeza/28,
0, 0, 15, 15,
15, 15, 20, 12,
20, 12, 17, 5,
17, 5, 17, -5,
17, -5, 20, -12,
20, -12, 15, -15,
15, -15, 0, 0/

```

#

```

data ojos/32,
10, 10, 10, 13,
10, 13, 13, 13,
13, 13, 13, 10,
13, 10, 10, 10,
10, -10, 10, -13,
10, -13, 13, -13,
13, -13, 13, -10,
13, -10, 10, -10/

```

#

```

data torax/36,
-52, 3, -25, 8,
-25, 8, -2, 4,
-2, 4, -2, -4,
-2, -4, -25, -8,
-25, -8, -52, -3,
-143, 0, -108, 12,
-108, 12, -52, 5,
-52, -5, -108, -12,
-108, -12, -143, 0/

```

#

```

data ptras/48,
-75, 8, -85, 12,
-85, 12, -86, 26,
-86, 26, -97, 40,
-97, 40, -130, 25,
-130, 25, -150, 35,
-150, 35, -152, 41,
-75, -8, -85, -12,
-85, -12, -86, -26,
-86, -26, -97, -40,
-97, -40, -130, -25,
-130, -25, -150, -35,
-150, -35, -152, -41/

```

#

```

data pdelan/32,
-60, 6, -70, 10,
-70, 10, -45, 30,
-45, 30, -75, 32,
-75, 32, -75, 38,
-60, -6, -70, -10,
-70, -10, -45, -30,
-45, -30, -75, -32,
-75, -32, -75, -38/

```

#

```

data pfron0/32,

```

```
-4, 4, 0, 9,
0, 9, -25, 12,
-25, 12, -43, 7,
-43, 7, -40, 5,
-4, -4, 0, -9,
0, -9, -25, -12,
-25, -12, -43, -7,
-43, -7, -40, -5/
```

```
#
data pfronl/8,
-4, 4, 0, 0,
-4, -4, 0, 0/
data icolor/5/, icojos/7/
data imagen/.false./
```

```
#
# dibuja el tronco
#
```

```
call dibpar(torax, fac, 0., icolor, imagen)
call dibpar(ptras, fac, 0., icolor, imagen)
call dibpar(pdelan, fac, 0., icolor, imagen)
if {ataq == 0}
  call dibpar(pfron0, fac, 0., icolor, imagen)
else
  {
    pfronl(4)=xat/fac
    pfronl(5)=(yat+2)/fac
    pfronl(8)=xat/fac
    pfronl(9)=(yat-2)/fac
    call dibpar(pfronl, fac, 0., icolor, imagen)
  }
  call dibpar(cabeza, fac, phi, icolor, imagen)
  call dibpar(ojos, fac, phi, icojos, imagen)
  return
end
```

```
*****
```

```
# SUBROUTINE DIBUJA PARTE DE CUERPO DE MANTIS
```

```
*****
```

```
subroutine dibpar(parte, scal, theta, icolor, imagen),
integer parte(1)
logical imagen
n=parte(1)
n=n+1
do i=2, n, 4
  {
    ix1=parte(i)
    iy1=parte(i+1)
    ix2=parte(i+2)
    iy2=parte(i+3)
    call rota(ifix(scal*ix1), ifix(scal*iy1), theta, i1, j1)
    call rota(ifix(scal*ix2), ifix(scal*iy2), theta, i2, j2)
    if(imagen)
      {
        call volteas(i1, j1, 1)
        call volteas(i2, j2, 1)
      }
  }
end
```

```

        call xline(i1,j1,i2,j2,icolor)
    }
    return
end
*****
#   FUNCION RND
*****
#
#   funcion que obtiene un numero random
#
#   ENTRADA:
#           X numero random obtenido anteriormente o la semilla
#           si es el primer acceso a esta funcion.
#
REAL FUNCTION RND(X)
common /esp/pi
i=x
x=x-i+pi
x=x**5
i=x
x=x-1
rnd=x
return
end

```


A P E N D I C E D .

B I B L I O T E C A S D E S O F T W A R E .

```

C*****
C****
C****          BIBLIOTECA      MANLIB1          ****
C****          ****
C*****
C
C
C*****
C**  subroutine mueve cursor          **
C*****
C
      subroutine muevec(x,y,z,ifun)
      integer x,y,z,p
      integer xmin,ymin,zmin
      integer xmax,ymax,zmax
      integer xpaso,ypaso,zpaso
      common/minim/xmin,ymin,zmin
      common/maxim/xmax,ymax,zmax
      common/increm/xpaso,ypaso,zpaso

C
C
      go to(100,200,300,400,500,600),ifun
25  continue
      call ring
      go to 1000

C-----
C Decremento en X
C-----
100  p=x-xpaso
      if(p.lt.xmin)go to 25
      x=p
      go to 1000

C-----
C Incremento en X
C-----
200  p=x+xpaso
      if(p.gt.xmax)go to 25
      x=p
      go to 1000

C-----
C Decremento en Y
C-----
300  p=y-ypaso
      if(p.lt.ymin)go to 25
      y=p
      go to 1000

C-----
C Incremento en Y
C-----
400  p=y+ypaso
      if(p.gt.ymax)go to 25
      y=p
      go to 1000

C-----
C Decremento en Z

```

```

C-----
500   p=z-zpaso
      if(p.lt.zmin)go to 25
      z=p
      go to 1000
C-----
C Incremento en Z
C-----
600   p=z+zpaso
      if(p.gt.zmax)go to 25
      z=p
      go to 1000
C
1000  continue
      return
      end
C
C
C*****
C**   subroutine convierte de 3 a 2 dimensiones   **
C*****
C
      subroutine d32(i,j,k,il,jl)
      common/proy/psi
      il=i+ifix(cos(psi)*j)
      jl=k+ifix(sin(psi)*j)
      return
      end
C
C
C*****
C**   subroutine rota cuerpo de                   **
C**           Mantis Religiosa                   **
C*****
C
      subroutine rota(x,z,theta,xr,yr)
      integer x,z,xr,yr
      real theta
      r=sqrt(float(x**2+z**2))
      alfa=0
      if(.not.(x.eq.0 .and. z .eq. 0))alfa=atan2(float(z),float(x))
      gamma=theta+alfa
      xr=r*cos(gamma)
      yr=r*sin(gamma)
      return
      end
C
C
C*****
C**   subroutine obten espejo de Cuerpo de       **
C**           Mantis Religiosa                   **
C*****
C
      subroutine voltea(xo,yo,ieje)
      integer xo,yo

```

```
      go to(100,200), leje  
C espejo respecto al eje X  
C  
100      continue  
        yo=-yo  
        go to 300  
C espejo respecto al eje Y  
C  
200      continue  
        xo=-xo  
300      continue  
        return  
        end
```

```

;*****
;***
;***          BIBLIOTECA  CRTLIB          ***
;***          *                ***
;*****
;

```

```

cdos      equ 5
          entry inkey,ring
inkey:    ld c,11          ;numero de llamada para test console ready
          call cdos       ;llama a rutina de cdos
          cp a,0          ;hay caracter
          jr z,final      ;salta si no hay
          ld c,128        ;si hay:numero de llamada para read char s/echo
          call cdos       ;llama a la rutina
final:    ld (hl),a       ;guarda caracter
          ret
;
;
ring:     ld c,2
          ld e,07H        ;bell char
          call cdos
          ret
          end

```

B I B L I O G R A F I A

[01] Balderrama, N. & Maldonado, H. (1971). Habituation of the deimatic response in the Mantid (STAGMATOPTERA BIOCELLATA). Journal of Comparative and Physiological Psychology.

[02] Maldonado, H. (1970). The deimatic reaction in the Praying mantis (STAGMATOPTERA BIOCELLATA). Z. Vergl. Physiologie.

[03] Maldonado, H. & redriguez, E. (1972). Depth Perception in the Praying Mantis. Physiology and Behavior. Vol. 8.

[04] Rilling, S., Mittelstaedt, H., Roeder, K.D. (1959). Prey Recognition in the Paying Mantis. Behavior XIV.

[05] Roeder, K.D. (1967). Nerve cells and insect behavior.

[07] Rossel, S. (1983). Binocular stereopsis in an insect. Nature, Vol. 302. No. 5911, pp. 821-822.

[08] Maldonado, H. (1972). A Learning process in the Praying Mantis. Physiology and Behavior. 9, 434-445. [09] Taro, Yamane. (1976). Statistics. An introductory analysis. 2nd. edition.

[09] Lara, R. La Cibernética del cerebro. En prensa.

[10] Piaget, J. (1975). Introducción a la Epistemología genética. Paidós, Argentina.

[11] Lara, R., Carmona, M., Daza, F. y Cruz, A. (1984). A global model of the neural mechanisms responsible for visuomotor coordination in toads. J. of theoretical Biology. En prensa.

[12] Lara, R., Franco, A., Salas, E. y Alcántara, V. (1984). Coordinación visuomotora en la Mantis Religiosa. Rev. Mex. de Ingeniería Biomédica. Vol. 5.