

# Presas y Cazadores: Un Enfoque desde las Redes Neurales

## Preys and Hunters: A Neural Networks Approach

José L. Uribe A., Ing., Adith B. Pérez O., Ing.

Departamento de Ingeniería de Sistemas, Pontificia Universidad Javeriana, Bogotá, Colombia

{uribej, adith.perez }@javeriana.edu.co

Recibido para revisión 28 de Noviembre de 2007, aceptado 14 de Febrero de 2008, versión final 28 de Febrero de 2008

**Resumen**—El presente artículo muestra el proceso de diseño e implementación de un Sistema Neural de toma de decisiones; haciendo énfasis en los subsistemas que lo conforman, como son el pre-procesamiento, procesamiento y post-procesamiento. De igual forma, pretende ser pionero para comenzar un proceso co-evolutivo respecto a las diferentes estrategias de Inteligencia Artificial que existen, sobre todo en los enfoques centralizados enfrentados a aproximaciones distribuidas, particularmente a través de Redes Neurales Artificiales y Sistemas MultiAgentes respectivamente.

**Palabras Clave**—Redes Neurales Artificiales, RNA, Sistemas Multi Agentes, SMA, Presas y Cazadores, Co-evolución, Simulación.

**Abstract**—This paper presents the design process and implementation of a system Neural decision-making; emphasizing on the subsystems that make up, such as pre-processing, processing and post-processing. Likewise, it aims to be a pioneer to begin a co-evolutionary process for different strategies Artificial Intelligence that exist, especially in the face of approaches centralized distributed approaches, particularly through Artificial Neural Networks and Systems MultiAgentes respectively.

**KeyWords**—Artificial Neural Networks, ANN, Multi Agent Systems, MAS, Preys and Hunter, Co-evolution simulation.

### I. INTRODUCCIÓN

La Inteligencia Artificial, de ahora en adelante IA, es un campo del conocimiento que puede tener dos focos de acción, en su ámbito de investigación.

El primer foco, busca estrategias que hagan que los artefactos creados por el hombre emulen el comportamiento del Cerebro Humano, en otras palabras que extrapolen el conocimiento que tienen y lo empleen en los momentos indicados y en el modo adecuado.

El segundo foco de investigación se relaciona con el aprovechamiento de las estrategias aceptadas como "patrones" o "métodos" de IA. Existen muchas clasificaciones para los diferentes métodos de IA, sin embargo una de las principales es la que se refiere a su nivel de "centralización", en donde obviamente se tiene dos grandes ramas, una centralizada y la otra distribuida.

En el Departamento de Ingeniería de Sistemas de la Pontificia Universidad Javeriana, se ha trabajado en el desarrollo de sistemas inteligentes distribuidos a través de Sistemas MultiAgentes, denominados de ahora en adelante (SMA), con diferentes modelos y estrategias cooperativas, destacándose principalmente las aplicaciones de Fútbol Cooperativo y Presas y Cazadores. En particular, los procesos de caza por parte de los depredadores es modelado a través de un SMA, pero las presas están desprovistas de toda inteligencia.

Es natural, desear "equilibrar la balanza" dado que el proceso de modelado por parte de los cazadores se realiza suponiendo a las presas totalmente azarosas, por lo que se hace interesante validar el modelo cuando "sus víctimas" tienen una posibilidad de escape o una mejor aproximación para decidir que dirección de salvamento deben tomar.

La intención del presente trabajo de investigación es empezar a realizar la "co-evolución tecnológica", pero desde otra perspectiva, desde un punto de vista más "bio-inspirado".

Si se está en un entorno de Presas y Cazadores, suponga que UD es una presa y que de un momento a otro lo acecha un cazador, UD por su instinto de supervivencia correrá en el sentido contrario al del cazador y tratará de buscar un lugar o dirección inaccesible por parte de su depredador.

La metáfora que se acaba de describir tiene varias connotaciones: Si UD analiza brevemente, la ruta de escape

seleccionada era la correcta, no obstante, en ningún momento pasó por su cabeza calcular o medir la velocidad lineal o angular del depredador, ni mucho menos se puso a crear un árbol de decisión, con cálculo de probabilidad incluido, para decidir su ruta de escape, sencillamente, UD. ya había "aprendido" que debía correr en sentido contrario, sin ejecutar ningún proceso algorítmico.

Por otra parte, y sin olvidarnos de la metáfora, a mediados del siglo pasado, la comunidad científica en general se encontraba en el dilema de ¿Cómo poder emular el comportamiento de la Inteligencia Humana?, y una de las respuestas es lo que se conoce como el Perceptrón, la Red Neural Artificial (de ahora en adelante se denominará RNA) más simple que existe.

Debido a las limitaciones inherentes del Perceptrón, esta estructura matemática evolucionó paulatinamente a redes perceptrónicas más complejas, hasta llegar a lo que se conoce hoy en día como los Perceptrones Multicapa. No obstante, entre los años 70's y 80's, cuando ya el "modelo multilayer" estaba plenamente definido, las potencialidades reales de este tipo de RNA no fueron explotadas, dado que no se contaba en ese entonces, con las capacidades de cómputo que se cuentan hoy en día para realizar los diversos procesos de entrenamiento, simulación y validación, que requieren los modelos neurales [1]

La mención de las RNA tiene mucho que ver con la metáfora descrita anteriormente, dado que el principio de funcionamiento (no se olvide que es una abstracción matemática) es similar a la descrita cuando UD. reaccionó contra su cazador, dado que no contiene ningún algoritmo o proceso complejo, simplemente responde de acuerdo a como se encuentren los pesos sinápticos que la conforman, de forma similar que UD responde cuando se le pregunta algo que ha aprendido, dado que cuando lo aprendió se modificaron conexiones en sus células nerviosas. [2][3]

Por tal motivo, y sin entrar en detalle del funcionamiento de las RNA, estas proveen soluciones que se denominan de computación suave, dado que en ningún momento se ejecuta un algoritmo estructurado o proceso semejante, por tal motivo, un Sistema Neural es un excelente candidato para responder a un problema "bio-inspirado" como lo es el de presas y cazadores, dado que su funcionalidad es totalmente "bio-inspirada".

Sin embargo, uno de los problemas más frecuentes que se tienen hoy en día, es no saber cómo construir un Modelo Neural, dado que se cuenta con mucha información de su funcionamiento, de los asombrosos resultados que ofrecen, pero en la mayoría de los casos, no mencionan cómo resolver un problema mediante RNA.

Con base en la problemática descrita, nace este artículo, para mostrar paso a paso una manera de resolver un problema no parametrizado, como lo es el de presas y cazadores, y cómo mediante un sistema de pre-procesamiento, procesamiento y post-procesamiento se puede en principio, proveer una ruta de escape a las presas en un entorno de presas y cazadores.

El presente artículo está dividido en 3 grandes secciones: en la primera, denominada de Modelado, se describirá todo el proceso para llegar a un Sistema Neural y solucionar el problema en particular que se está enfrentando, el cuál es tomar la decisión de una ruta de escape para las presas. En la segunda sección, se describe la implementación, en la tercera se analizará el desempeño del Sistema Neural y por último una sección de conclusiones y trabajos futuros.

## II. MODELADO

Antes de continuar, es necesario precisar que el entorno de presas y cazadores que se menciona en el presente artículo se refiere exclusivamente a entidades modeladas por medio de software. Tanto la implementación del simulador, como las diversas técnicas de IA descritas en este documento están basadas en el framework BESA. [5][6][7]

El Sistema Neural a implementar en cada presa está dividido en 3 partes:

*Sistema de pre-procesamiento*, que como su nombre lo indica se encarga de pre-procesar la información que va a llegar a la red. Es una etapa de acondicionamiento en la cual las variables de entrada al sistema se restringen a un conjunto de valores analizables por el sistema de procesamiento.

El *sistema de procesamiento*, tras previa adecuación de las entradas al sistema completo, es el encargado de tomar las decisiones más importantes, es el cerebro central del sistema y por ende el módulo más importante.

Por último, el *sistema post-procesamiento*, es el módulo encargado de convertir las respuestas del sistema de procesamiento en variables que entienda el mundo, en otras palabras debe convertir las decisiones neurales en movimiento, dirección, velocidad de escape de la presa etc.

Antes de continuar y con base en la definición de Sistema de Pre-Procesamiento, las presas y los cazadores son móviles que cuentan con un arreglo de sensores cada uno. Para las diversas pruebas que se realizan, las presas cuentan con tres sensores, uno simulando el sentido de la "vista" y los otros dos, simulando dos oídos. Los cazadores, solo cuentan con un sensor, el cual pueden mover libremente.

De esta forma, las presas y los cazadores, pueden conocer el estado del mundo cuando se está ejecutando la simulación. En la Figura 1 pueden observarse los móviles que corresponden a los agentes simulados.

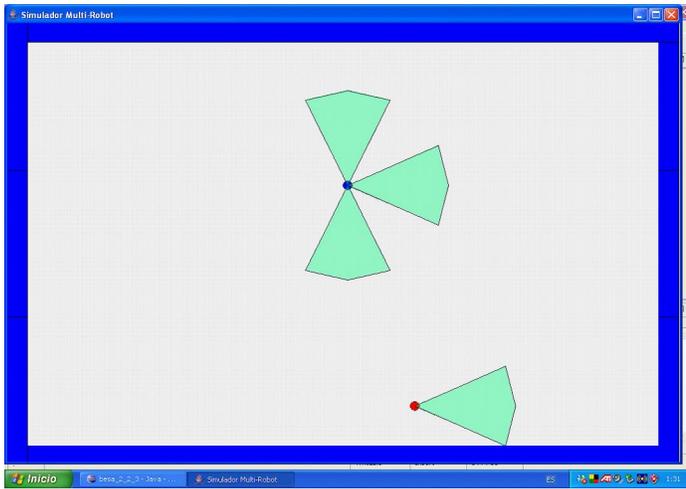


Figura 1. Ilustración de los sensores de los móviles. Cazadores en rojo y Presas en azul.

Con base en lo anterior, se define el siguiente conjunto de variables respecto a la percepción que tiene la presa:

- UCFR: Único Cazador al Frente.
- PCFR: Pocos Cazadores al Frente.
- MCFR: Muchos Cazadores al Frente.
  
- UCDER: Único Cazador a la Derecha.
- PCDER: Pocos Cazadores a la Derecha.
- MCDER: Muchos Cazadores a la Derecha.
  
- UCIZQ: Único Cazador a la Izquierda.
- PCIZQ: Pocos Cazadores a la Izquierda.
- MCIZQ: Muchos Cazadores a la Izquierda.

Si se analiza la descripción de las variables mencionadas, se tiene un componente lingüístico en sus definiciones, tal como se hace en los sistemas difusos. No obstante si se revisan concienzudamente las variables, se está haciendo referencia a la cantidad de cazadores percibidos, por lo tanto la naturaleza de este término es discreta, dado que es imposible ver 1.6 cazadores a la izquierda o 0.5 al frente.

La aclaración que se hace es importante, porque no se está hablando de la proximidad del cazador, que puede convertirse en una variable difusa real dado que la distancia es fácilmente expresable en términos lingüísticos (Por ejemplo: muy lejos, lejos y cerca) adicionalmente a su naturaleza continua, sino de la cantidad de cazadores vistos u oídos.

Por esta razón se pueden declarar umbrales muy bien definidos, para la cantidad de cazadores percibidos y que ayudarán a especificar mejor las variables antes descritas:

- Se considera que se tiene un único cazador por sensor (sentido - vista ó audición -), cuando solamente es posible detectar a un solo depredador.
- Se considera que se tiene pocos cazadores por sensor, cuando se tiene 2 o 3 cazadores sentidos.
- Se considera que se tienen muchos cazadores por sensor, cuando se tengan más de 3 cazadores sentidos.

Nótese que los anteriores umbrales pueden llegar a ser modificados, pero eso depende del criterio con el que se esté desarrollando y evaluando el modelo de escape.

Con lo dicho anteriormente, se tiene entonces que la naturaleza de las variables descritas posee un trasfondo booleano, porque describen el comportamiento de cada sentido de la presa con base en lo que están sensando y le asignan un criterio de verdad. Por ejemplo:

Si la presa está "viendo" un solo cazador al frente y no "escucha" ninguno al lado derecho y "escucha" 5 al lado izquierdo, puede concluir lo que muestra la Tabla No 1.

TABLA 1. EJEMPLO DEL USO DE LAS VARIABLES DEL SISTEMA NEURAL

Proposición	Variable	Criterio de Verdad
“Veo UN solo cazador al frente”	UCFR	Verdadero
“Veo POCOS cazadores al frente”	PCFR	Falso
“Veo MUCHOS cazadores al frente”	MCFR	Falso
“Escucho UN solo cazador a la izquierda”	UCIZQ	Falso
“Escucho POCOS cazadores a la izquierda”	PCIZQ	Falso
“Escucho MUCHOS cazadores a la izquierda”	MCIZQ	Verdadero
“Escucho UN solo cazador a la derecha”	UCDER	Falso
“Escucho POCOS cazadores a la derecha”	PCDER	Falso
“Escucho MUCHOS cazadores a la derecha”	MCDER	Falso

Por lo tanto, se puede concluir que las entradas al Sistema Neural (procesamiento), son variables binarias y que están en función de los sensores de las presas, es decir en lo que éstas pueden "ver" y "oír".

Ya definidas las variables de entrada al Sistema Neural, es necesario especificar cuáles y cómo van a ser las variables de salida. Al recordar que este es un modelo de Presas y Cazadores y que al ser un modelo posee varias restricciones, se debe enfatizar una en particular, la cual es que no se cuentan con patrones de vitalidad o de "desgaste físico", en otras palabras, los cazadores y las presas no necesitan fuentes de energía y ninguno de ellos se desgasta (pierde energía) en la ejecución de sus actividades.

Por esta razón, el sistema de toma de decisiones de las presas solo analizará que dirección debe tomar la presa para no ser cazada, dado que siempre se estará moviendo con su velocidad máxima. Este parámetro es modificado por el usuario en los archivos de configuración de la simulación.

Entendida entonces la variable de salida del sistema como la dirección que debe tomar la presa cuando "se siente atacada", es necesario identificar su naturaleza y "rango". Con el fin de proponer una solución clara que permita evaluar y validar el modelo propuesto, se decidió discretizar la variable de salida en 8 valores, como se muestra a continuación:

TABLA 2.  
 CUANTIFICACIÓN DE LA VARIABLE DE SALIDA DE DIRECCIÓN QUE DEBE TOMAR LA PRESA

Dirección que puede tomar la presa	Grados
Este (E)	0
Noreste (NE)	45
Norte (N)	90
Noroeste (NO)	135
Oeste (O)	180
Suroeste (SO)	225
Sur (S)	270
Sureste (SE)	315

Comprendiendo que el proceso de diseño del Sistema Neural va más allá del "calculó" de la RNA, se presenta a continuación un Diagrama de Bloques en donde está la interacción de cada uno de los sub-bloques, así como el dominio de las variables de entrada y de salida de cada uno.

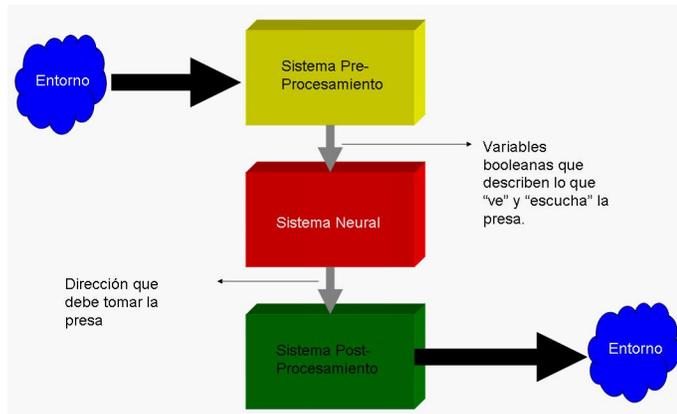


Figura 2. Diagrama de Bloques del sistema de escape de las presas

El sistema de pre-procesamiento recoge la información del entorno y con base en los sensores de la presa, genera un conjunto de variables booleanas que indican que tan "acechada" se encuentra. Esta información es transmitida al Sistema Neural el cual debe indicar que dirección debe tomar para evitar ser cazada. Gracias a la discretización de la variable de salida, está será codificada de manera binaria para facilitar así el trabajo de entrenamiento de las redes neurales. Posteriormente la decisión es llevada al sistema de post-procesamiento para que decodifique la dirección seleccionada (no olvide que esta se encuentra en formato binario cuando sale del sistema de procesamiento - neural) y la convierta en información que reconozca el entorno.

Gracias a que se cuenta con un conjunto de variables de entrada y salida, es necesario realizar una abstracción adicional que permita ingresar información al Sistema Neural y de la misma forma definir cómo será la salida del sistema, que como se mencionó será en formato binario.

Por lo anterior, es necesario definir unos "patrones de conducta" de escape y de esta forma, lograr entrenar al Sistema Neural.

Para ello, considérese la siguiente situación en el entorno de presas y cazadores:

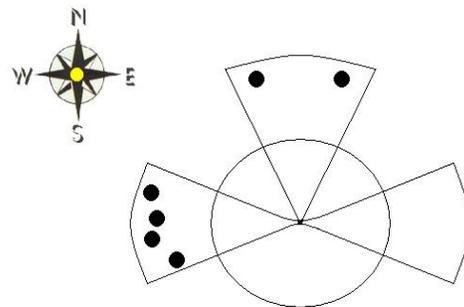


Figura 3. Situación de "enfrentamiento" entre presas y cazadores.

Destáquese que las dimensiones mostradas entre presas y cazadores no corresponden al real, se tiene un detalle más amplio de la presa para ilustrar de mejor manera la situación. En este ejemplo, la presa "ve" 2 cazadores al frente, "escucha" 4 al lado izquierdo y "no ve ni escucha" nada al lado derecho.

En esta situación en particular, se puede ver claramente que la presa debe escapar en dirección sureste (SE), por tal motivo el sistema de procesamiento debe indicarle a la presa que debe escoger esta dirección. Como se ha dicho anteriormente, estas decisiones se han codificado de manera binaria, dado que se tienen 9 opciones (8 direcciones y una de no cambio de la dirección que lleva), es necesario tener 4 bits o dígitos binarios, como se muestra a continuación:

TABLA 3.  
CODIFICACIÓN BINARIA DE LA VARIABLE DE SALIDA DE DIRECCIÓN QUE DEBE TOMAR LA PRESA

Dirección que puede tomar la presa	Grados	Codificación
No cambiar dirección	--	0000
Este (E)	0	1000
Noreste (NE)	45	0001
Norte (N)	90	0010
Noroeste (NO)	135	0011
Oeste (O)	180	0100
Suroeste (SO)	225	0101
Sur (S)	270	0110
Sureste (SE)	315	0111

El siguiente diagrama, sintetiza lo dicho anteriormente:

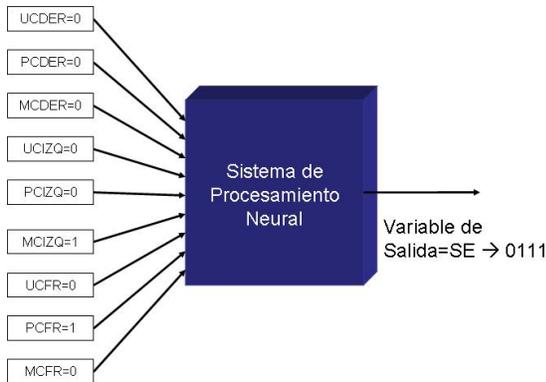


Figura 4. Ilustración de la dinámica de las variables de entrada y salida al Sistema Neural

Si se analiza detalladamente se tiene entonces un conjunto de 9 variables de entrada booleanas, que podría tomar  $2^9=512$ , no obstante y si se observa con cuidado dentro de las combinaciones posibles, pueden aparecer  $UCDER=1$ ;  $PCDER=1$ ;  $MCDER=1$  y las otras variables igualadas a cero, por tanto, semánticamente hablando se tiene que al mismo tiempo la presa esta viendo un cazador a la derecha y pocos (2 o 3) cazadores a la derecha y muchos (mas de 3) a la derecha, lo que es imposible. De esta forma es necesario crear un conjunto de patrones relevantes para entrenar a la red neural y de esta forma evaluar el conocimiento adquirido y su capacidad de generalización.

Al descartar situaciones como la descrita anteriormente, se cuenta con un conjunto de 64 escenas relevantes para realizar el entrenamiento neural.

Conforme se ha estructurado el problema, se tiene que no es linealmente separable y por lo tanto es necesario realizar una implementación multicapa para poder realizar el entrenamiento. No obstante, esta decisión implica una selección de cuantas capas ocultas y cuantas neuronas debe tener cada capa oculta. Para poder tomar estas decisiones, se cuenta con algunas "reglas de oro", que proporcionan ciertas heurísticas para el diseño neural.

Para resolver el número de capas ocultas, se citará a Martín del Río [3]:

"Una arquitectura de características similares a las del Perceptrón Multicapa, con una única capa oculta, resultaba ser un aproximador universal de funciones, no necesariamente booleanas. No obstante, se pueden recurrir a dos capas ocultas, pero esto se debe a cuestiones prácticas."

Dado que no existe un proceso formal para hallar el número de neuronas en la capa oculta, se tomarán dos criterios, el piramidal y el "ponderado" de entradas y salidas.

El criterio piramidal descrito por Martín del Río[3], señala que la RNA tenga un aspecto de pirámide, en otras palabras, el número de neuronas en cada capa va decreciendo desde las entradas a las salidas.

El criterio "ponderado", descrito por Warren [4] señala: "Para calcular el número de nodos ocultos nosotros usamos una regla general:  $(\text{Número de entradas} + \text{salidas}) * (2/3)$ ". En el sistema, se tienen 9 entradas y 4 salidas, lo que al aplicar la fórmula anterior da como resultado 8.66, pero siguiendo el criterio piramidal, se redondeará por debajo, es decir se tomarán 8 neuronas ocultas. De esta forma se obtiene Sistema Neural de (9-8-4), como se muestra a continuación:

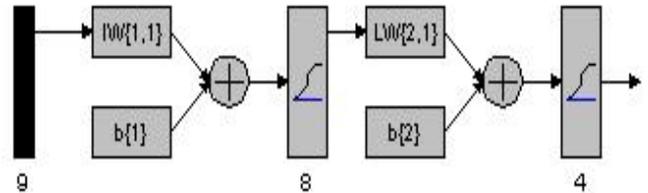


Figura 5. Implementación Neural. Perceptrón Multicapa (9-8-4)

Ya con el Sistema Neural construido, se procede a realizar el entrenamiento con las 64 escenas relevantes mencionadas (Ver Anexo).

### III. IMPLEMENTACION

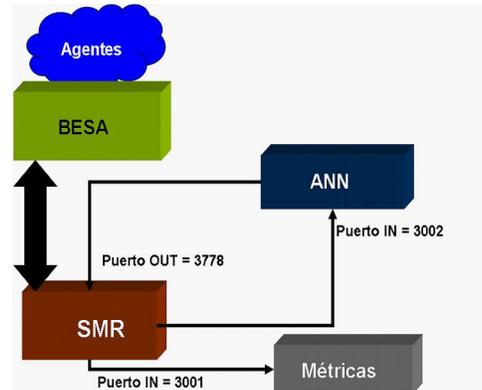


Figura 6. Diagrama de Bloques de los sistemas que interactúan con los procesos de simulación

Ya realizado el modelo neural, es necesario recordar uno de los objetivos fundamentales de este artículo, el cual es ser pionero en la co-evolución tecnológica en cuanto a desarrollos de IA. Por tal motivo, es necesario enfrentar el modelo neural descrito anteriormente a un modelo inteligente, en este caso frente a un SMA, es decir presas neurales frente cazadores multiagentes.

Con base en lo anterior, y dado que se debe proveer la funcionalidad de interacción entre varios móviles, es necesario contar con un Simulador MultiRobot (SMR). De forma adicional, como los cazadores implementan su estrategia de inteligencia de forma cooperativa a través de SMA, es necesario contar con un framework que permita la interacción entre el mundo de los objetos móviles y el de los agentes, el cual es BESA [5][6][7]. Es muy importante mencionar que ambos proyectos han sido llevados a cabo por el Departamento de Ingeniería de Sistemas de la Pontificia Universidad Javeriana, y que se han ido perfeccionando de acuerdo a las nuevas necesidades que han surgido.

Se debe recalcar que en el Diagrama de Bloques de la Figura No 6 la aplicación referenciada como ANN, la cual es la implementación del Sistema Neural, se comunica con el SMR por medio de dos puertos. El No 3002, le permite conocer el estado del mundo al sistema, para que tome la decisión correcta. El puerto No 3778, establece una comunicación entre el Sistema Neural y el SMR, para que ya tomada la decisión a que ángulo debe girar la presa, se la pase al SMR y efectivamente mueva el objeto móvil.

De igual manera, se tiene la aplicación "Métricas", la cual por medio de otro puerto de comunicación, recibe del SMR mensajes de cuando existen eventos especiales, en este caso en particular, cuando existe un proceso de caza.

Con la descripción anterior, se verifica la modularidad que presentan las distintas aplicaciones, dado que no fue necesario realizar cambios en la implementación ni en BESA, ni en el SMR.

**IV. PRUEBAS**

Las variables de control más importantes que se definieron en el protocolo de pruebas son:

- Número de Agentes en cada simulación: 16.
- Tamaño del Entorno: 1000x700 píxeles.
- Duración de las pruebas: 180 segundos.
- Relación Presas/Cazadores = 25/75; 50/50; 75/25.
- Distancia focal de los sensores de las presas: 150 y 220 píxeles.

Para validar el modelo neural, se realizaron más de 800 pruebas, destacándose los siguientes experimentos:

*Primer Experimento:* Simulación de presas azarasas (desprovistas de inteligencia) vs. Cazadores azarasos en dos máquinas con diferentes capacidades de cómputo:

Características de la Máquina M1:

- **M1:** PC Estándar, Procesador: Intel® Core™ 2 Duo Processor de 2.4 GHz, Memoria RAM: 1 GByte, Sistema Operativo: Windows XP Service Pack 2. Virtual Machine: JRE Versión 6 Update 1.

- √ Parámetros de Simulación: -Relación (P/C)=50/50  
 -Tamaño de Sensores = Normal  
 -Presas y Cazadores sin Inteligencia

- **M2:** Laptop, Procesador: Intel® Centrino de 2.6 GHz, Memoria RAM: 512 MByte, Sistema Operativo: Windows XP Service Pack 2. Virtual Machine: JRE Versión 6 Update 1.

- √ Parámetros de Simulación: -Relación (P/C)=50/50  
 -Tamaño de Sensores = Grande  
 -Presas y Cazadores sin Inteligencia

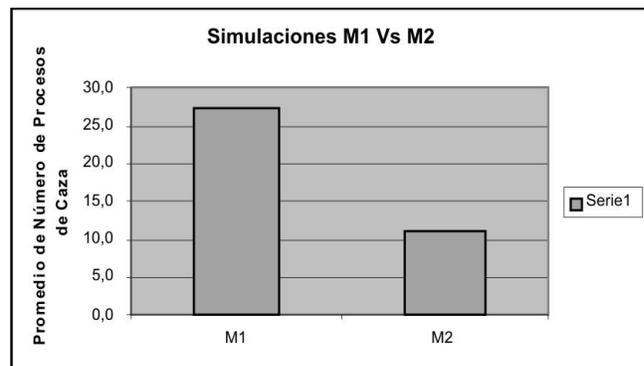


Figura 7. Diagrama de Barras de los promedios de procesos de caza en M1 y M2

*Segundo Experimento:*

Comparación entre las Presas Azarasas (no-inteligentes), Sistema Neural implementado en las presas y cazadores azarasos. Se promedian los "números de procesos de caza".

TABLA 4. TABLA COMPARATIVA ENTRE PRESAS AZARASAS, NEURALES Y CAZADORES AZAROSOS

	Relación P/C = 25/75		Relación P/C = 50/50		Relación P/C = 75/25	
	Presas Azarasas vs. Cazadores Azarasos	Presas Neurales Defensivas vs. Cazadores Azarasos	Presas Azarasas vs. Cazadores Azarasos	Presas Neurales Defensivas vs. Cazadores Azarasos	Presas Azarasas vs. Cazadores Azarasos	Presas Neurales Defensivas vs. Cazadores Azarasos
<b>Promedio</b>	22,20	9,00	27,70	13,60	17,60	8,00
<b>Desviación Estándar</b>	4,90	3,60	6,50	5,00	5,60	3,90
<b>Porcentaje de Reducción de "cazas"</b>	60%		51%		54%	

*Tercer Experimento:*

Comparación entre el Sistema Neural implementado en las presas y el SMA de los cazadores. Se promedian los "números de procesos de caza".

TABLA 5.  
TABLA COMPARATIVA CAZADORES SMA VS. PRESAS RNA.

	Relación P/C=25/75	Relación P/C=50/50	Relación P/C=75/25
	RNA vs. SMA	RNA vs. SMA	RNA vs. SMA
Promedio	9,5	13,3	11,7
Desviación Estándar	3,4	5,5	3,5

## V. ANALISIS, CONCLUSIONES Y TRABAJO FUTURO

Con base en la Figura No 7, se puede observar una reducción muy importante del número de procesos de caza, solamente cambiando las características de los computadores en donde se realizan las pruebas, lo cual es un resultado inesperado dado que el aumento del tamaño de los sensores no cambia en lo absoluto el comportamiento del sistema en este momento, puesto que ninguno de los entes procesa la información que percibe, dado que tanto presas como cazadores tienen un comportamiento azaroso.

Por lo tanto, una conclusión importante es que en el momento de desarrollar este tipo de pruebas, es recomendable tener en cuenta una variable que influye mucho en los resultados y que hasta este momento no se ha tenido en cuenta y es la "capacidad de cómputo" en las máquinas sobre las cuales se ejecutan las simulaciones, dado que como se puede observar en el caso descrito se pueden presentar errores o los llamados "falsos positivos".

Con base en los resultados de la Tabla No 4 y 5, el resultado para el Modelo Neural es positivo, dado que fue el resultado de ajustes y análisis que hicieran "competitiva" la respuesta, puesto que tiene un comportamiento muy similar cuando los cazadores son azarosos o cooperativos, lo que implica que es una solución robusta, debido a que al modificar el entorno, como colocar inteligencia a sus "depredadores", el resultado no sufre grandes cambios, lo que demuestra que la Red Neural aprende y funciona adecuadamente, y que por lo tanto los objetivos del proyecto se han cumplido.

Adicionalmente, nótese que la reducción de los procesos de caza es mayor al 50% para las tres relaciones planteadas, lo que confirma aún más la validez del Modelo Neural de escape.

Ya con la validez del Modelo Neural comprobada para este contexto de presas y cazadores, es importante mirar el proceso co-evolutivo de este artículo con otra perspectiva. Gracias a la abstracción de variables booleanas mencionadas en el Modelado (Sección II), el problema que plantea este artículo se hubiera podido resolver con una ecuación lógica o una tabla de búsqueda.

No obstante, si se adoptarán soluciones como las que se acaban de mencionar, el sistema no sería fácilmente escalable en cuanto a la parametrización del entorno, dado que una variable más (booleana o real) cambiaría todo el modelo, no se le podría dar de forma natural una prioridad a un conjunto de variables y sobre todo, se estaría describiendo un proceso reactivo y no inteligente.

Por tal motivo, esta solución neural, presenta un caso base exitoso que abarca muchos otros, gracias a la capacidad intrínseca de las RNA de generalizar el modelo creado. Por ejemplo, sin cambiar el entrenamiento actual, el Sistema Neural creado, es capaz de absorber el ruido de los sensores. De igual forma, en el caso de tener varios sensores independientes pero que apunten a la misma dirección, por ejemplo no un sentido de vista, sino "dos ojos", el modelo neural no debe reentrenarse para seguir obteniendo un conjunto de resultados favorables.

De esta forma y ya para terminar, el gran aporte de este proyecto es incentivar y propender por una co-evolución entre los diversos trabajos de investigación sobre IA que se realicen. En este caso, se busca "equilibrar la balanza" hacia el lado de las presas para que no fueran desprovistas de inteligencia. Por ejemplo: Puede ser interesante para futuros Trabajos de Investigación, perfeccionar las estrategias de cooperación entre los cazadores, dado que ya cuentan con un rival más preparado y cuando se llegue a ese escalón, hacer que las presas tengan otras técnicas de supervivencia más que el escape, como por ejemplo: mimetismo, o llegada a territorios inaccesibles por parte de los cazadores etc.

## REFERENCIAS

- [1] Hagan Martin, Neural network design. Boston, Massachusetts : PWS, 1996
- [2] Haykin Simon, Neural networks: a comprehensive foundation, New Jersey: Printice Hall, 1994
- [3] Martín del Río Bonifacio, Sanz Alfredo. Redes Neuronales y Sistemas Difusos. Ed Alfaomega. Segunda Edición.2001.
- [4] Sarle Warren. Servicio FTP "Neural Network FAQ". Disponible en Línea: <ftp://ftp.sas.com/pub/neural/FAQ.html>. Consultado el 7 de Abril de 2007.
- [5] E. González, C. Bustacara and J. Avila. BESA: Behavior Oriented, Event driven Social Based Agent Framework
- [6] E. González, V. Arteaga, S. Romero, C. Bustacara. RoboCoop: A MultiRobot Control Agent Architecture
- [7] E. González, A. Vásquez, B. Plata, A. Pérez, C. Bustacara. CCS: Commit Based Cooperation Strategy for MultiRobot Systems.

**Uribe J.** Es ingeniero Electrónico egresado de la Pontificia Universidad Javeriana, Actualmente es candidato a grado como ingeniero de Sistemas de la misma universidad y está desarrollando una Maestría en Ingeniería de Sistemas en la Universidad de los Andes. El ingeniero Uribe se ha destacado por investigar temas donde convergen sus dos profesiones.

**Perez A.** es Ingeniero de Sistemas egresado de la Pontificia Universidad Javeriana; actualmente se desempeña como profesor de tiempo completo del departamento de Ingeniería de Sistemas de la misma universidad, ha estado encargado de las cátedras de Sistemas Operativos, Sistemas Basados en el Conocimiento, Estructuras de Datos, Programación Orientada a Objetos además de coordinar las Practicas Profesionales y Proyectos de

Grado. El profesor Pérez se encuentra realizando su Proyecto de grado de Maestría en el área de Inteligencia Artificial concentrándose en cooperación y aprendizaje en Sistemas Multi-Agentes.

**ANEXOS**

A continuación, se presentan los 64 casos bases con lo que se desarrolló el entrenamiento neural:

