

Sergio A. Rojas

Johanna Ramírez

Óscar Romero

## RESUMEN

Los animales artificiales son agentes inteligentes de software y/o hardware que demuestran un comportamiento similar al de los animales reales. Para esto deben contar con un mecanismo de aprendizaje que les permita adquirir nueva información mientras interactúan con su entorno, de forma que puedan utilizarla para mejorar su medida de desempeño. Los Sistemas Clasificadores son una técnica de aprendizaje maquina basada en un enfoque de computación evolutiva que puede ser utilizada con este fin. En este artículo presentamos los resultados de la simulación de un animal artificial que aprende a comportarse en un mundo incierto utilizando dos tipos de Sistemas Clasificadores.

**Palabras clave:** aprendizaje por refuerzo, sistemas clasificadores, animats.

### A Classifier System-based Artificial Animal

Artificial animals are hardware/software intelligent agents that must demonstrate a similar behavior like natural animals. They should have some kind of learning mechanism to acquire new information while they are interacting with their environments, so they can improve their fitness evaluation. Classifier Systems, which are a machine learning technique based on an evolutionary approach, can be applied for this purpose. In this paper we present some results of an artificial animal simulation that use two different kinds of Classifier Systems.

**Key words:** reinforcement learning, classifier systems, animats.

## I. INTRODUCCIÓN

Los Sistemas Clasificadores (SC) propuestos por Holland [1], son una técnica de aprendizaje de máquina que utiliza una representación del conocimiento utilizando reglas simples llamadas *clasificadores*. Estos son los encargados de dirigir el comportamiento del sistema dentro de su ambiente. Un SC puede aprender a clasificar la información de entrada estableciendo reglas generales (clasificadores) encargados de responder a situaciones similares. Así como en cualquier sistema de control, el SC usa la retroalimentación enviada por su entorno para adaptar sus clasificadores de forma que el sistema responda mejor. Los SC poseen algunas ventajas en compara-

ción con otro tipo de sistemas basados en reglas, como los Sistemas Expertos (SE):

- Los SE requieren una representación rígida y exhaustiva del conocimiento a través de las reglas, mientras el SC emplea un módulo de aprendizaje guiado por un Algoritmo Genético (AG) para descubrir nuevas reglas que se adapten mejor a las necesidades de su ambiente.
- Las reglas de un SE son deducidas por humanos, mientras que las del SC las deduce (o las refina) el mismo sistema.
- Cuando una regla de un SE no cumple ninguna funcionalidad no es retirada de la base de reglas, mientras que en el SC esto se logra mediante el reemplazo por nuevos clasificadores.
- En un SE ocurren conflictos cuando existen varias reglas aplicables a una situación, recurriendo a mecanismos de tratamiento de la incertidumbre, para resolverlos. En los SC se resuelven de manera implícita por medio de un algoritmo de distribución de créditos.

Como se puede observar, los SC tienen características deseables en un sistema de aprendizaje, como su capacidad de adaptación a diversos escenarios, de trabajar con información incompleta y de clasificar las situaciones por jerarquías. En particular, pueden ser útiles como mecanismo de aprendizaje en agentes inteligentes. En este artículo se presenta un experimento realizado con un tipo de agentes inteligentes [2] conocida como animales artificiales o *animats* (ver [3]). En la siguiente sección se hace una revisión de los fundamentos de un SC. En la sección 3 se describe con detalle el animat y el experimento. La sección 4 presenta los resultados encontrados, y en la última sección se discuten algunas conclusiones.

## II. SISTEMAS CLASIFICADORES

El SC se comunica con su ambiente mediante unos detectores que le permiten capturar información de entrada, y unos efectores que ejecutan la siguiente acción determinada por las reglas. Internamente, el SC cuenta con un almacén de clasificadores, un mecanismo de descubrimiento de reglas, una casa de apuestas, un sistema de distribución de ganancias, y un sistema de aprendizaje basado en AG (ver fig. 1).

El SC se comunica con su ambiente mediante unos detectores que le permiten capturar información de entrada, y unos efectores que ejecutan la siguiente acción determinada por las reglas.

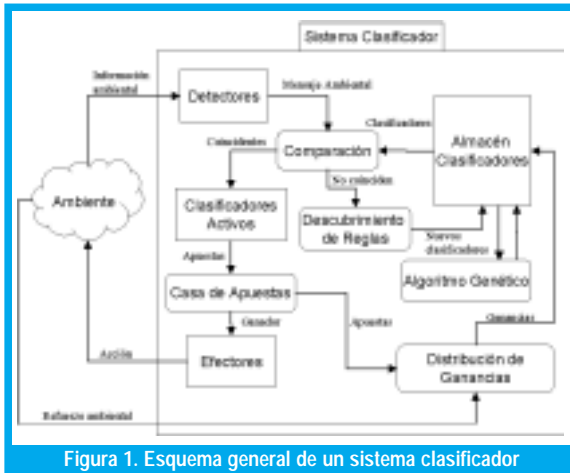


Figura 1. Esquema general de un sistema clasificador

### 2.1. Almacén de clasificadores y descubrimiento de reglas

En un SC tradicional, la información de entrada y salida se representa mediante señales booleanas. Por ejemplo, un SC para el control de la electricidad en una casa, podía tener como entradas binarias variables que indiquen si ya son más de las 6 de la tarde, si la temperatura es inferior a los 5° C, y si hay presencia de personas en la casa; dependiendo del valor de estas variables las salidas podrían ser encender o apagar los bombillos, o la calefacción, o programar la alarma de la puerta principal. Así pues, la información de entrada debe ser procesada por el SC. Esta es la tarea de los clasificadores. Una vez los detectores capturan los datos, lo envían como un mensaje a un almacén de clasificadores con el que cuenta el sistema. Cada clasificador tiene la siguiente estructura:

<clasificador> :: <condición> ? <mensaje> / <g>

La parte <condición> es un arreglo de dígitos que pertenece a al alfabeto {0, 1, #}. Cada dígito representa una variable de entrada; el carácter '#' es un comodín que indica que no importa si la variable vale '1' o '0'. La parte <mensaje> es un arreglo binario que representa la salida sugerida por ese clasificador. Adicionalmente, cada clasificador cuenta con un valor de ganancia <g> que indica la relevancia que tiene el clasificador para el sistema.

Cada vez que entra un mensaje enviado por los detectores, el sistema busca en su almacén de clasificadores aquellos cuya <condición> coincida. Estos son los que tienen prioridad para emitir una salida. El sistema escoge uno de ellos, y envía ese <mensaje> a los efectores. En algunos casos, el <mensaje> puede ser reenviado al almacén de clasificadores (como si el clasificador ganador estuviera “pidiendo ayuda” a otros). Cuando el mensaje ambiental no coincide con ninguno de los clasificadores, se crearía uno nuevo, con una <condición> que coincida, y con un <mensaje> aleatorio (a esto se le llama el mecanismo de descubrimiento de reglas). Retomando el ejemplo anterior, si el sistema detecta que son las 7 pm, y no hay nadie en casa (sin importar la tempe-

ratura), entonces programará la alarma sin encender las luces ni la calefacción (fig. 2a); pero si la situación es que hay una persona en casa, a las 4 de la tarde, con una temperatura de 1° C, entonces encenderá la calefacción (fig. 2b).

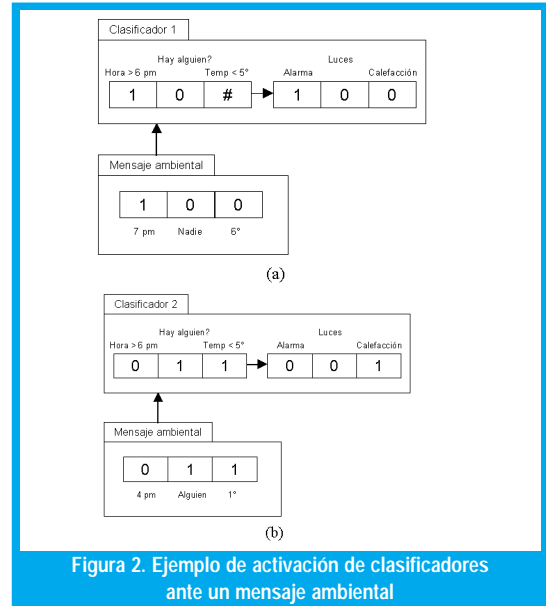


Figura 2. Ejemplo de activación de clasificadores ante un mensaje ambiental

### 2.2. La casa de apuestas y la distribución de ganancias

Como varios clasificadores pueden coincidir con el mensaje detectado, se realiza una subasta para escoger el clasificador ganador. Esta subasta la lleva a cabo la casa de apuestas. Básicamente, cada uno de los clasificadores activados hace una apuesta proporcional a su nivel de ganancia, para tratar de ganar el derecho a enviar su <mensaje>. La casa de apuestas entonces decide cual es el ganador.

El aprendizaje se realiza gracias a la retroalimentación recibida por el entorno. Cuando el mensaje ganador se ejecuta a través de los efectores, el ambiente se encarga de evaluar la acción efectuada mediante alguna función de desempeño. Como resultado reenvía una señal de refuerzo y este valor ingresa al sistema, y se toma como una ganancia que debe ser repartida entre los clasificadores ganadores (los que ganaron la apuesta) y aquellos que contribuyeron a su activación (porque enviaron mensajes internos). Este sistema de distribución de ganancias permite que los clasificadores manejen una especie de “economía digital” en la que van ganando y perdiendo capital dependiendo del desempeño del sistema. Esta forma de aprendizaje, mediante premios y castigos se denomina *aprendizaje por refuerzo* [4].

La dinámica de la ganancia de cada clasificador dependerá entonces de las apuestas que realice, del refuerzo recibido y distribuido por la casa de apuestas, además de un impuesto que se le cobra por permanecer en sistema. La ecuación (1) muestra la regla de modificación de ganancias del clasificador en función del tiempo.

La dinámica de la ganancia de cada clasificador dependerá entonces de las apuestas que realice, del refuerzo recibido y distribuido por la casa de apuestas, además de un impuesto que se le cobra por permanecer en sistema.

Un animat es un sistema que debe funcionar en tiempo real, realizando las acciones correctas, tomando decisiones en el momento preciso.

$$S_i(t+1) = S_i(t) - B_i(t) - T_i(t) + R_i(t) \quad (1)$$

donde  $S_i$  es la ganancia del clasificador,  $B_i$  es la apuesta realizada,  $T_i$  es el impuesto cobrado por la casa de apuestas y  $R_i$  es el refuerzo recibido desde el ambiente.

Usualmente, la apuesta es un valor proporcional a la ganancia total del clasificador, aumentado con un valor proporcional a su especificidad<sup>1</sup> y para que la escogencia no sea tan determinística, se le puede añadir un ruido aleatorio con distribución normal. En suma, la apuesta se define como en (2),

$$B_i(t) = C_{apu} * S_i(t) + C_{esp} * Especificidad + N(\sigma_{apu}) \quad (2)$$

donde  $C_{apu}$  y  $C_{esp}$  son la proporción de la apuesta y la especificidad respectivamente, y  $N(\sigma_{apu})$  es la función de ruido. Por su parte, el impuesto se calcula como una proporción de las ganancias, como se ve en (3),

$$T_i(t) = C_{impuesto} * S_i \quad (3)$$

### 2.3. El Algoritmo Genético

Finalmente, los SC simulan un proceso evolutivo artificial en el que las diferentes especies (o grupos) de clasificadores se reproducen y compiten entre sí, con el fin de perfeccionarse y crear clasificadores cada vez mejores a medida que pasan las generaciones. Se trata de un AG que se ejecuta por épocas y que mejora sustancialmente la calidad del aprendizaje obtenido por el sistema.

En el caso del SC, el AG ejecuta durante varias generaciones operaciones de selección de individuos, cruce sexual y mutación, de la manera tradicional [5]. La única diferencia radica en la operación de mutación, en la que se escoge uno de los genes del cromosoma (o sea del clasificador) y aleatoriamente se intercambia por cualquiera de los otros dos posibles (si es un '1' se puede mutar equiprobablemente por un '0' o por un '#'). Los nuevos clasificadores obtenidos, se reemplazan por los clasificadores más mediocres del almacén, es decir, aquellos que poseen menor ganancia. El AG se invoca cada cierto número de iteraciones, especialmente cuando el sistema no encuentra clasificadores aptos para responder a mensajes ambientales.

### 2.4. Sistema Clasificador Simple y Sistema Cognitivo-1

Los dos sistemas fueron propuestos por Holland [1]. El SC Simple (SCS) implementa la estructura y los algoritmos de distribución de ganancias descritos anteriormente. Por su parte el Sistema Cognitivo-1 (CS-1, por sus siglas en inglés) tiene un mecanismo distinto para la repartición de ganancias. La prin-

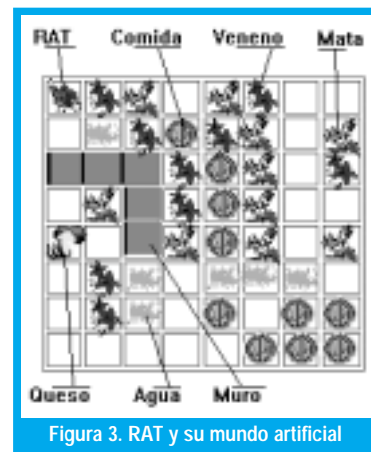
<sup>1</sup> Se puede definir la especificidad como la cercanía o parecido que tiene la <condición> del clasificador al mensaje ambiental. Recuérdese que como el comodín '#' puede coincidir con '0' o '1', entre menos comodines tenga y más dígitos concuerden exactamente, el clasificador es más específico, y por lo tanto se puede pensar, más confiable para responder al mensaje.

cipal diferencia radica en que mientras que en el SCS el refuerzo recibido es repartido de manera inmediata por la casa de apuestas, en el CS-1 estas ganancias se acumulan durante cierto periodo de tiempo (época) al cabo del cual se distribuyen entre todos los clasificadores que participaron en él. Además, en el CS-1 la estructura de los clasificadores se divide en porciones para manejar los mensajes ambientales, los mensajes internos, y los mensajes recientemente enviados. La ganancia de los clasificadores depende tanto del refuerzo recibido, como de la edad del clasificador (el tiempo que ha permanecido en el almacén sin sufrir modificaciones en su estructura) y la frecuencia de activación (la cantidad de veces que se activa durante una época).

Con el fin de medir el desempeño de estos dos mecanismos en simulaciones de aprendizaje maquina, hemos implementado un animal artificial que aprende a comportarse en un ambiente incierto. El experimento se describe en la siguiente sección. Los resultados se presentan en la sección 4.

## III. DESCRIPCIÓN DEL EXPERIMENTO

Un animat es un sistema que debe funcionar en tiempo real, realizando las acciones correctas, tomando decisiones en el momento preciso basadas en las señales enviadas hacia y desde el ambiente, en su estructura fisiológica (interna) y en las consecuencias de su comportamiento actual y futuro. El animat debe querer algo, tener un fundamento motivacional.



En nuestro experimento, se diseñó el animat "RAT" que debe resolver el problema de búsqueda de comida dentro de un mundo desconocido para él. El ambiente artificial es una rejilla bidimensional dividida en 64

cuadros. Cada casilla puede estar ocupada por el animat, por comida, por agua, por veneno o por un muro. La tarea de RAT será aprender a moverse dentro de su ambiente evitando obstáculos y venenos, y procurando buscar alimento y agua. Su meta final es conseguir el queso.

### 3.1. Diseño del clasificador

RAT tiene capacidad para sentir los elementos que están a su alrededor (en las 8 casillas vecinas a su posición). Las señales que puede percibir se muestran en la tabla 1.

TABLA 1. SEÑALES QUE PUEDE PERCIBIR RAT

Objeto	Letra	Código	Significado	Refuerzo
Muro	M	001	obstáculo con el que RAT puede golpearse	Negativo
Blanco	B	010	casilla sin ocupar, RAT se puede mover allí	Ninguno
Agua	A	011	cada gota de agua satisface la sed de RAT	Positivo
Comida	C	100	cada pedazo de comida satisface el hambre de RAT	Positivo
Queso	Q	101	objetivo final; comerlo significa un premio grande	Positivo
Mata mala	MM	110	aunque RAT puede comerla, le causa indigestión	Negativo
Mata veneno	MV	111	le causa una indigestión más severa	Negativo

Por otra parte, las acciones que puede ejecutar RAT, se refieren a su movimiento. Se puede mover hacia cualquiera de las casillas vecinas a su posición actual. Estas acciones se resumen en la tabla 2.

TABLA 2. REPERTORIO DE ACCIONES DE RAT

Acción	Letra	Código	Significado
Noroeste	NO	000	moverse a la casilla ubicada al noroeste de la actual
Norte	N	001	moverse a la casilla ubicada al norte de la actual
Noreste	NE	010	moverse a la casilla ubicada al noreste de la actual
Este	E	011	moverse a la casilla ubicada al este de la actual
Sureste	SE	100	moverse a la casilla ubicada al sureste de la actual
Sur	S	101	moverse a la casilla ubicada al sur de la actual
Suroeste	SO	110	moverse a la casilla ubicada al suroeste de la actual
Oeste	O	111	moverse a la casilla ubicada al oeste de la actual

La información capturada por RAT se agrupa en una cadena que constituye el mensaje ambiental; se construye iniciando con el objeto detectado en la casilla norte, y recorriendo los demás vecinos en el sentido de las manecillas del reloj; por ejemplo, si RAT estuviera en la siguiente situación:

M	Q	C
C	RAT	A
MM	MV	

Percibiría el siguiente mensaje ambiental: <Q C A B MV MM C M>, ante el cual debería moverse hacia el norte para comerse el queso. Un buen clasificador sería entonces:

<Q C A B MV MM C M> ? <N>

lo que en binario se codificaría como:

<101 100 011 010 111 110 100 001> ? <001>

Finalmente, se han diseñado 3 escenarios diferentes para experimentar el aprendizaje de RAT. Cada uno de ellos se describe en la tabla 3.

TABLA 3. LOS MUNDOS DONDE PUEDE HABITAR RAT

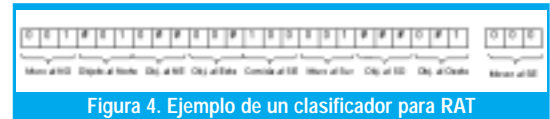
Escenario	Letra	Código	Significado
Línea recta	L	00	Camino sin obstáculos, hay un rastro de comida que lleva hasta el queso
Semicircunferencia	S	01	Solo existe un camino que lleva al queso en forma circular, y esta rodeado por muros
Camino amenazante	C	10	Existen varias posibilidades para llegar al queso, con amenazas como matas, veneno y muros

De esta forma, la <condición> de los clasificadores está compuesta por 26 bits (24 de los vecinos y 2 para el escenario), y el <mensaje> por 3 bits.

A modo de ejemplo, ante la siguiente situación

M	Q	C
C	RAT	A
MM	MV	

RAT debería tener por lo menos un clasificador que le indique que debe moverse hacia el sureste, para obtener la comida, evitando estrellarse con los muros (como el que se observa en la fig. 4).



Obsérvese la utilidad de los comodines; gracias a ellos se pueden formar esquemas que representan más de un estado del problema. En el ejemplo anterior hay 8 comodines, luego el clasificador representa  $2^8 = 256$  estados del problema, ya que cada '#' puede ser sustituido por un '1' o un '0'. Por lo tanto, la subcadena "0#1" (objeto al oeste) puede representar tanto un muro ("001") como una gota de agua ("011"). Así el animat puede aprender a generalizar, como en este caso, que al tener comida al sureste (SE), su acción será moverse en esa dirección, sin importar si tiene un muro o una gota de agua al oeste.

Como se puede ver, las acciones indicadas por el <mensaje> de cada clasificador pueden no ser las óptimas. El animat debe ir afinando sus clasificadores para expresar un mejor comportamiento. Para ello se utiliza el algoritmo de distribución de créditos, que va modificando las ganancias de cada clasificador en razón a los refuerzos recibidos (calmar el hambre, indigestarse, comerse el queso, etc.), y el AG, que mediante el cruce y la mutación de clasificadores puede descubrir mejores configuraciones para las cadenas correspondientes a la <condición> y el <mensaje> de la población de clasificadores.

### 3.2. Diseño de los escenarios

Como se mencionó anteriormente, en el experimento se utilizaron tres ambientes con diferentes grados de complejidad. El objetivo era comprobar que RAT era capaz de adaptarse a los tres, a pesar de que representaran ambientes completamente distin-

tos. El ambiente L se puede observar en la figura 5. Este ambiente no tiene obstáculos, y contiene un rastro de comida que guía al animat hacia el queso. Se debe recordar que cuando RAT se coloca en su ambiente, desconoce por completo sus características, y debe aprender a moverse de acuerdo a los refuerzos recibidos por sus acciones exploratorias. En este mundo se espera que RAT no pierda tiempo moviéndose por las regiones vacías, sino que siga el patrón de comida hasta llegar al queso. El comportamiento óptimo demoraría 8 pasos, si la posición inicial del animat es la esquina superior izquierda.

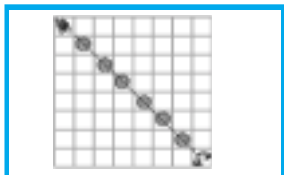


Figura 5. Aspecto del Ambiente L

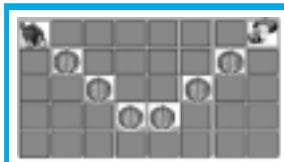


Figura 6. Aspecto del Ambiente S

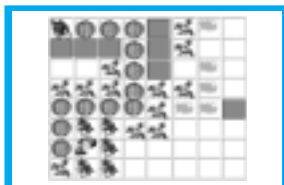


Figura 7. Aspecto del Ambiente C

El ambiente S se diseñó como una especie de semicircunferencia, como se ve en la figura 6. En este caso, RAT debe descubrir que se encuentra rodeado de muros, y que cualquier movimiento que intente, diferente a ir donde encuentra comida, le representa un castigo. Así, se ve obligado a tener que aprender el único camino disponible.

Por último, el ambiente C (fig. 7) se diseñó como el de mayor grado de dificultad. En este caso, RAT estará sujeto a

experimentar con nuevos elementos en su hábitat, como las matas malas, las matas venenosas y los depósitos de agua. Se espera que el animat siga el camino de comida que lo lleva directamente al queso empleando 13 pasos como máximo, evitando desviarse o comer matas.

### 3.3. Parámetros de ejecución

La población de clasificadores inicial se genera aleatoriamente y a cada clasificador se le asigna una ganancia inicial de 500 unidades. Los demás parámetros del SC, incluyendo los del AG se muestran en la tabla 4.

TABLA 4. PARÁMETROS DE EJECUCIÓN DEL SC Y DEL AG

Variante	Valor	Descripción
$C_{apu}$	0.1	10% de la ganancia del clasificador
$C_{esp}$	0.5	Tasa especificidad
$S_{apu}$	1.0	Desviación estándar para la función de ruido
$S_0$	500	Ganancia inicial para todos los clasificadores
$n$	1000	Tamaño del almacén de clasificadores
$C_{imp}$	0.034	Tasa de impuesto
$e$	20	Frecuencia de iteraciones para la ejecución del AG
$R_p$	10%	Tasa de selección
$P_m$	50%	Tasa de mutación
$P_c$	100%	Tasa de cruce sexual

## IV. RESULTADOS OBTENIDOS

Con el fin de medir los resultados que demuestran el aprendizaje de RAT, se realizaron diez experimentos en los tres escenarios descritos anteriormente. Cada vez que se iniciaba un experimento, la población de clasificadores se creaba de manera aleatoria. Los resultados aquí mostrados, corresponden al promedio obtenido para cada ambiente. En cada experimento, se observaron varios periodos de aprendizaje, que consistían en ubicar al animat en su posición inicial y permitir que se moviera en su mundo hasta que llegara a la meta (hasta que se comiera el queso).

### 4.1 Resultados en el ambiente L

En la figura 8 se observa los resultados del comportamiento del RAT. Se realizaron tres tipos de experimentos diferentes: el primero con un mecanismo aleatorio de selección de movimientos, el segundo con el SCS sin AG, y un tercero con AG. Solo con este último el animat tiene una curva de aprendizaje que converge al óptimo (8 pasos para comer el queso). Esto demuestra que la sola dinámica de apuestas y distribución de ganancias del SCS no es suficiente para lograr un buen aprendizaje; el AG es importante para que el animat pueda afinar su almacén de clasificadores de forma que resuelva el problema exitosamente. Es importante resaltar que el aprendizaje es bastante rápido (se alcanza casi en 7 periodos de aprendizaje), si se tiene en cuenta que al principio el animat no “sabe” que la comida como las frutas y el queso le retribuyen un beneficio; esto lo descubre por medio de la experiencia y la retroalimentación que recibe del ambiente.

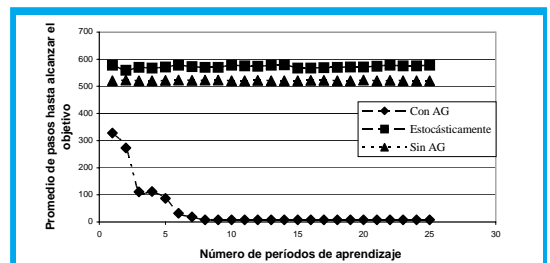


Figura 8. Aprendizaje de RAT en ambiente L utilizando SCS

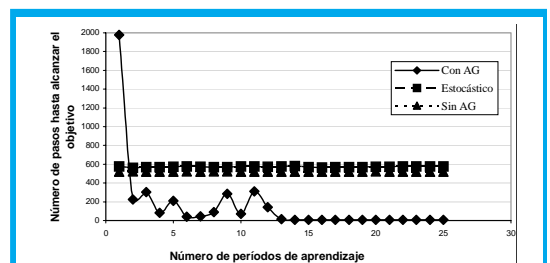


Figura 9. Aprendizaje de RAT en ambiente L utilizando CS-1

Cuando RAT utiliza un CS-1, el comportamiento es similar, pero el aprendizaje es más demorado, como se observa en la figura 9. De nuevo es necesario utilizar el AG para que la curva converja, pero se demora cerca de 15 periodos de aprendizaje en en-

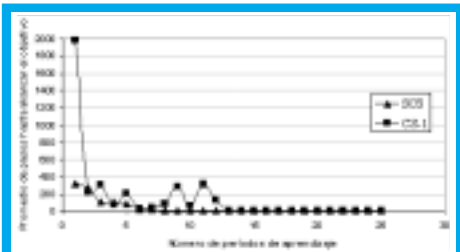


Figura 10. Comparación entre el SCS y el CS-1

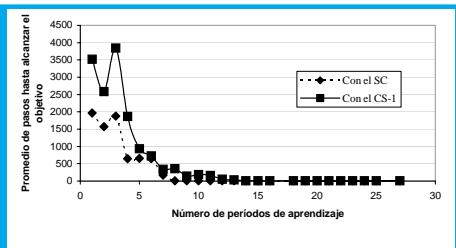


Figura 11. Aprendizaje de RAT en ambiente S con SCS y CS-1

Sin embargo, el tiempo total para aprender la estrategia óptima es más demorado (entre 8 y 9 periodos de aprendizaje). Esto es debido a que el animat debe adaptar al nuevo ambiente los clasificadores que había aprendido en su experiencia pasada: la primera parte del recorrido era similar a la del escenario anterior, pero después tuvo que aprender a desplazarse diagonalmente hacia arriba para llegar a la meta.

### 4.3 Resultados en el ambiente C

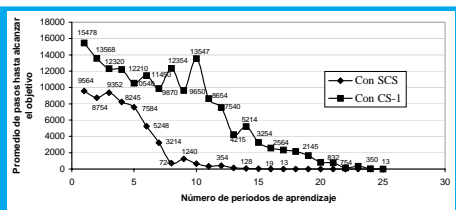


Figura 12. Aprendizaje de RAT en ambiente C con SCS y CS-1

Este ambiente es un poco más complejo que los dos anteriores, ya que son muy escasos los movimientos similares a los aprendidos en los ambientes 1 y 2. Por tal motivo RAT tiene que reorganizar completamente su almacén de clasificadores para adaptarse a su nueva situación. Con el SCS, el comportamiento de RAT convergió a la ruta más corta (13 pasos) en el período de aprendizaje número 16, mientras que con el CS1 convergió en el período 25. En este caso se observa un desempeño mucho mejor del SCS (ver fig. 12).

Adicionalmente, se pudo observar que RAT empleó conocimiento adquirido en experiencias anteriores para aplicarlo a la situación actual, y reducir a 12 el número de pasos necesarios para comer el queso. Esta situación se ilustra en la figura 13. La parte a) muestra el comportamiento que se esperaba aprender (13 pasos). En la parte b) se muestra como al desplazarse en diagonal, el animat disminuyó el número de pasos a 12. Este movimiento diagonal lo había aprendido principalmente en el ambiente L y reforzado en el ambiente C.

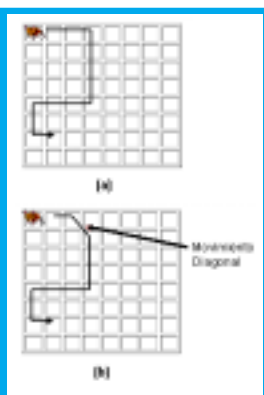


Figura 13. RAT usando conocimiento adquirido (a) Comportamiento esperado de 13 pasos (b) Disminución a 12 pasos

contrar la ruta óptima. En la figura 10 se observa el rendimiento comparativo de RAT con SCS y con CS-1 después del quinto periodo de aprendizaje. Indudablemente aprende mejor cuando utiliza el SCS.

### 4.2 Resultados en el ambiente S

RAT fue ubicado en este ambiente después de haberse entrenado en el ambiente L. La curva de aprendizaje se observa en la figura 11.

En este caso el desempeño es similar en los dos sistemas, aunque con el SCS, RAT requiere menos pasos para encontrar el queso en los primeros periodos.

## V. DISCUSIÓN

Al observar los resultados de este experimento, podemos concluir que los Sistemas Clasificadores son herramientas efectivas para la implementación de sistemas de aprendizaje maquina, ya que permiten que se desarrollen capacidades para la adquisición de nueva información, y la adaptación de conocimiento adquirido, para ser aplicado en situaciones amenazantes que inclusive nunca antes hayan sido enfrentadas por el sistema. Más aún, el sistema alcanza a cumplir sus metas eficientemente, y hasta puede descubrir soluciones inesperadas que pueden ser calificadas como inteligentes. Otra característica importante es que la convergencia en el aprendizaje es rápida, al igual que la asimilación de un comportamiento más adecuado al ambiente en particular.

Este tipo de sistemas de aprendizaje por refuerzo se convierten en una alternativa viable para el desarrollo de agentes inteligentes que puedan aprovechar la información de su entorno para mejorar su efectividad. En particular, dentro del grupo ACME se está trabajando en un proyecto de informática educativa para desarrollar un tutor inteligente para la enseñanza de aritmética básica en educación primaria [6]. El SC podría aplicarse para lograr que el tutor descubriera algunas características del estilo cognitivo de su usuario al interactuar con el por medio de preguntas, y pudiera personalizar su contenido para optimizar su proceso de enseñanza. Este tipo de aplicaciones, así como la experimentación con otras herramientas de aprendizaje serán motivo de nuestro trabajo futuro.

## VI. REFERENCIAS BIBLIOGRÁFICAS

- [1] Holland, J.H. (1953). *Induction, Processes of Inference, Learning and Discovery*. Michigan: Addison-Wesley.
- [2] Rojas, S.A. (2001). "El e-AlterEgo: Fantasmas en Internet". En: *Revista Ingeniería*, año 2001, semestre 1. Facultad Ingeniería, Universidad Distrital Francisco José de Caldas.
- [3] Rojas, S.A. (1998). *Disertación Teórica sobre Simulaciones Inspiradas Biológicamente para el Estudio del Comportamiento Adaptativo*. Monografía de grado. Facultad de Ingeniería, Universidad Nacional de Colombia.
- [4] Sutton, R.S.; Barto, A.G. (1998). *Reinforcement Learning: An Introduction*. The MIT Press.
- [5] Goldberg, D.E. (1986). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Cambridge: MIT Press.
- [6] Romero, O.; Ramírez, J. (2002). *Tutor inteligente para la enseñanza operaciones básicas de aritmética en alumnos de educación básica primaria*. Trabajo de grado. Facultad de Ingeniería, Universidad Distrital Francisco José de Caldas. [Autores](#)

### Sergio A. Rojas

Ing. de Sistemas, U. Nacional de Colombia. Especialista en Ing. de Software, U. Distrital. Profesor/Investigador de la Facultad de Ingeniería, Universidad Distrital FJC. Director del Grupo de Interés en Adaptación, Computación & Mente (ACME-UD) [srojas@udistrital.edu.co](mailto:srojas@udistrital.edu.co)

### Johanna Ramírez

Estudiante de Ing. de Sistemas, U. Distrital Francisco José de Caldas. Investigadora del Grupo de Interés en Adaptación, Computación & Mente (ACME-UD) [johanna\\_rc@hotmail.com](mailto:johanna_rc@hotmail.com)

### Oscar Ramírez

Estudiante de Ing. de Sistemas, U. Distrital Francisco José de Caldas. Investigador del Grupo de Interés en Adaptación, Computación & Mente (ACME-UD) [ojrlopez@hotmail.com](mailto:ojrlopez@hotmail.com)