

## **Sistemas inmunes artificiales aplicados a la robótica: agarre de objetos**

Daniel Ramirez, Jesús Ortiz, Pedro Ponce, Arturo Molina

Instituto Tecnológico y de Estudios Superiores de Monterrey, Ciudad de México,  
México

dan.ramirez@itesm.mx, jesuss031@gmail.com, pedro.ponce@itesm.mx, armolina@itesm.mx

**Resumen.** Este documento expone un nuevo algoritmo basado en el Sistema Inmune Natural. Dicho algoritmo está diseñado para producir nuevo conocimiento en forma de anticuerpos a partir de ejemplos limitados en cantidad los cuales toman el papel de antígenos. El algoritmo y la teoría son explicados, además de la aplicación donde se probó su funcionamiento. La aplicación consiste en un guante de datos que permite dar ejemplos a una mano robótica. Los resultados se presentan de tal manera que permita corroborar la funcionalidad del sistema y del algoritmo al realizar la tarea de agarre. Finalmente se presenta una propuesta de embeber en el algoritmo un sistema de reconocimiento de formas mediante la clusterización de datos obtenidos por los sensores integrados al sistema para posteriormente clasificarlos.

**Palabras clave:** manipulación de objetos, robótica, inteligencia artificial aplicada, sistemas inmunes artificiales, computación evolutiva.

### **Artificial Immune Systems Applied to Robotics: Grip of Objects**

**Abstract:** This document exposes a new Artificial Immune System (AIS) algorithm, inspired in the Natural Immune System (NIS) and certain characteristics. The algorithm presented emulates certain procedures and utilizes concepts such as learning from demonstration and reinforcement learning. The algorithm is designed to produce new knowledge in form of antibodies from limited data samples as antigens. The algorithm and the theory involved is explained in this document, as well as the application chosen, which is a humanoid hand enabled to sense force and flexion in fingers in order to reproduce grasping postures. Results are presented in

order to confirm the functionality of the algorithm and the success at grasping objects.

**Keywords:** objects manipulation, robotics, artificial intelligence, artificial immune systems, evolutionary computation.

## 1. Introducción

El aprendizaje maquina es un área de la inteligencia artificial que durante años ha encontrado inspiración en la naturaleza para reproducir ciertos fenómenos o sistemas de manera computacional, con el fin de resolver diversos problemas y desarrollar aplicaciones [1]. Las áreas de aplicación son tan amplias y van desde aplicaciones relacionadas con la internet y el análisis de datos en dicha plataforma [2], [3], casas y edificios inteligentes [4], [5] hasta cultivos de alimentos con sistemas inteligentes [6], [7]. No es sorpresa que el Sistema Inmune Natural (SIN) atrajera la atención de diversos investigadores alrededor del mundo para de alguna manera reproducir ciertas de sus características inherentes de tal manera que fuera posible codificarlas y aplicarlas a diversos problemas.

El SIN es un ejemplo de un Sistema de aprendizaje adaptivo que no solo se adapta durante toda la vida de su huésped también durante toda su existencia crea una librería o repertorio de anticuerpos misma. Esta librería es utilizada para defender el cuerpo del huésped de una manera más rápida cuando por segunda ocasión encuentran al mismo antígeno. Así la respuesta será mucho más rápida y efectiva que en el primer encuentro. La tarea que tiene encomendada el SIN no es una fácil e involucra un gran número de agentes dentro de los cuales podemos destacar; células blancas de la sangre, células madre, nodos linfáticos, al timo entre otros.

La aportación de esta investigación es el desarrollo de un algoritmo basado en el SIN, emulando ciertos procedimientos para poder aplicarse a la robótica, específicamente al agarre de objetos. Este algoritmo pretende ser sencillo comparado con algunos otros que usan retroalimentación visual, nubes de puntos 3D o la cinemática inversa de una mano.

Al algoritmo se le ha nombrado CODA por su acrónimo en inglés Cognición a partir de datos (Cognition from Data), y para poder probarlo se ha construido un Sistema completo que consta de un guante de datos que recaba información de las posiciones y fuera de cada dedo en la mano del humano para poder usar esto como ejemplo y pasarlo al algoritmo para su posterior procesamiento. Otra parte del sistema consiste en una mano humanoide que se encargará de reproducir las poses de agarre para el objeto a partir del simple ejemplo obtenido por el guante y el procesamiento del algoritmo.

## 2. La teoría detrás de CODA

### 2.1. Sistema inmune artificial

El primer artículo que describió la importancia del Sistema Inmune Natural para la comunidad estudiantil del aprendizaje máquina fue escrito en 1986 por J. Doyne Farmer [8], donde el SIN es visto como un clasificador capaz de eliminar material externo al sistema original. Después de este artículo no fue hasta que Forrest et al. [9] en 1994 describiera en su trabajo una aplicación real del algoritmo llamado “Selección negativa”. La lista de algoritmos continúa con el Algoritmo de selección clonal o CLONALG que ha sido usado para reconocimiento de patrones y optimización de funciones multimodales con resultados positivos [10].

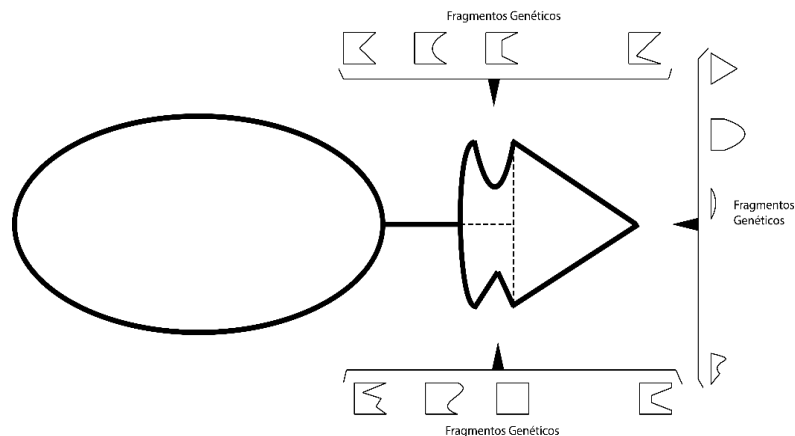
Otros algoritmos relacionados al SIN son; Redes Inmunológicas, células dendríticas y algunos derivados de los algoritmos mencionados. Para un mejor estudio de dichos temas los libros de De Castro y Timmis [11] así como el de Das Gupta [12] deben ser un excelente punto de comienzo.

La tabla 1, resume aquellas características importantes que se pueden encontrar en sus modelos computacionales del SIN. De esta tabla se han extraído características para el desarrollo del algoritmo presentado en este documento, sin embargo, es importante entender que el modelo del SIN es mucho más complejo y que los modelos computacionales han sido simplificados para poder ser emulados. Para un estudio más completo del SIN es necesario referirse a [13] de donde la tabla 1 fue extraída. Algunos de los comportamientos más importantes del SIN que se han imitado y que en el algoritmo CODA están presentes, son los siguientes.

**Tabla 1.** Características del Sistema Natural Inmune que lo hacen una herramienta tan ponderosa del cuerpo humano. Las características aquí presentadas son emuladas por muchos de los algoritmos llamados Sistemas Inmunes Artificiales.

CARACTERÍSTICA	IMPORTANCIA FUNCIONAL
<b>ESPECIFICIDAD</b>	Asegura que los distintos antígenos provoquen respuestas específicas
<b>DIVERSIDAD</b>	Permite al Sistema inmune responder a una gran variedad de antígenos
<b>MEMORIA</b>	Conduce a una mejora en las respuestas a exposiciones repetidas de un mismo antígeno
<b>EXPANSIÓN CLONAL</b>	Aumenta el número de antígenos específicos para mantener el ritmo de reproducción de los microbios
<b>ESPECIALIZACIÓN</b>	Genera respuestas que son óptimas para la defensa en contra de diferentes tipos de microbios
<b>CONTRACCIÓN Y HOMEOSTASIS</b>	Permite al Sistema inmune responder a antígenos completamente nuevos sin previos encuentros
<b>NO REACTIVIDAD A PROPIOS</b>	Previene daño al portador del sistema durante ataques a elementos foráneos

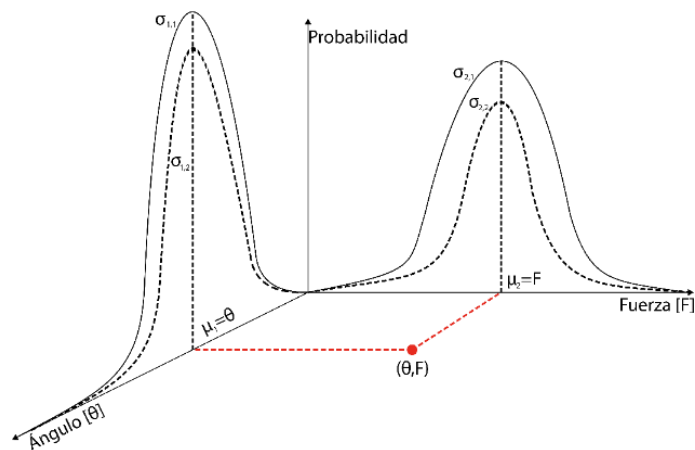
**Generación de diversidad:** El SIN es capaz de producir su propia diversidad a partir de sus células y de información que estas obtienen. El mecanismo que usa es recombinar de manera aleatoria fragmentos de genes en cada célula B nueva para así poder obtener nuevos anticuerpos. Este proceso, simple pero efectivo es capaz de producir anticuerpos muy diversos de manera masiva y ha funcionado durante mucho tiempo. Este proceso, el cual se puede apreciar en la figura 1, es capaz de producir anticuerpos para antígenos incluso si anteriormente no se había tenido contacto con el patógeno portador dentro del cuerpo humano.



**Fig. 1.** Ejemplo simple de cómo es creado un anticuerpo a partir de mezclas aleatorias de diversos genes.

En el algoritmo CODA, el proceso es emulado de la siguiente manera. Primero se obtiene el antígeno, el cual es extraído por el sistema de aprendizaje por demostración que consiste en el guante de datos del cual se hablará en secciones posteriores. Toda la información obtenida por este sistema es el patógeno y cada uno de los datos es un antígeno, información que es entregada al proceso de clonación/mutación del algoritmo para de esta manera generar la diversidad para producir nuevos y diferentes datos (anticuerpos).

Los ejemplos obtenidos son separados del arreglo y tratados según la variable a la que pertenezcan (posición y fuerza) a partir de ese momento cada dato es usado como la media  $\mu$  y encima de cada uno de ellos una distribución normal es fijada de tal manera que se obtengan anticuerpos dentro de esa distribución con una desviación estándar  $\sigma$ , proceso ilustrado por la figura no. 2. Esto quiere decir que en un problema en el cual se tuviesen 3 dimensiones o variables a medir, se tendrían tres distribuciones normales una para cada variable. Tomar el ejemplo como la media de la distribución hará que todas las variaciones se hagan según este número y de esta manera se asegura que no haya datos incongruentes que pudieran estar fuera de la distribución y que no servirían para nada.



**Fig. 2.** Diferentes densidades de probabilidad para cada dimensión del patógeno son creadas durante el procedimiento de clonación y de mutación. La letra griega sirve para definir la variable posición y la letra F la fuerza.

De la figura 2 el parámetro  $\sigma$  muestra diversos subíndices, que se leen de la siguiente manera:

$$\sigma_{x,y}; \text{ para todas las dimensiones, y por cada posible } \sigma; x, y \in \mathbb{Z}.$$

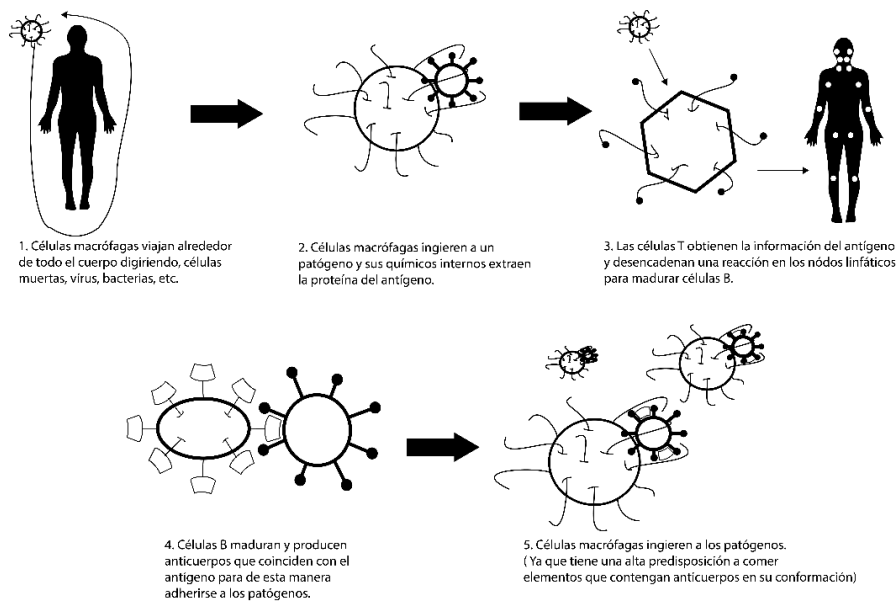
En la aplicación que se expone en este documento solamente se usan dos dimensiones como puede observarse en la figura anterior y esto es debido a la configuración de los sensores en el sistema y las variables que se desean extraer.

**Selección negativa:** Este procedimiento es de suma importancia ya que elimina a todas aquellas mutaciones con desventaja (medidas en comparación a un parámetro en el caso del algoritmo CODA una función de recompensa) que se produjeron al generar diversidad, este procedimiento también es conocido como apoptosis. Debido a que el proceso de generación de diversidad es aleatorio es posible que se creen anticuerpos que contengan datos erróneos y que puedan ser dañinos para el huésped, incluso que puedan atacar células del huésped por error, es por eso que este proceso de eliminación de este tipo de anticuerpos con desventajas es tan importante.

**Selección clonal:** Desde que el proceso de generación de diversidad es activado alrededor del cuerpo humano. En los nodos linfáticos habrá una gran cantidad de células B que comenzaran a producir anticuerpos con información errónea para esos casos la selección negativa se activa. Sin embargo, habrá células B que comiencen a crear anticuerpos funcionales y con alto potencial, es por eso que estas células B son alentadas a clonarse para de esta forma producir una mayor cantidad de anticuerpos funcionales. Este

procedimiento es importante ya que la es por medio de la hipermutacion somática que las células B producen cambios finos en los anticuerpos para crear los más capaces.

La figura 3 muestra un resumen de los procesos que se han explicado con anterioridad, y fue el modelo computacional simplificado que utiliza el algoritmo CODA.



**Fig. 3.** El sistema inmune adaptivo y sus procedimientos de defensa al encontrarse con un patógeno [14].

En el algoritmo CODA selección negativa y la selección clonal, están unidas a partir de la implementación de una función de evaluación que busca a los anticuerpos más capaces de reproducir la tarea deseada. La función de recompensa para esta aplicación está diseñada para poder obtener el mejor rendimiento del algoritmo basándose en la tarea a realizar. Es por esto que fuerza y posición son las variables escogidas, ya que permiten caracterizar una posición para manipular el objeto fácilmente. Las implementaciones de dichas funciones de recompensa se encuentran explicadas con mayor detalle en [14].

¿Por qué es importante evaluar la recompensa esperada para cada anticuerpo? Imagine que el sistema ha obtenido un mal ejemplo durante el proceso de aprendizaje por demostración, esto pudiera ser por datos corruptos o por falla en algún sensor. Esto provocaría que la variable  $\mu$  se colocará en una región del plano donde produciría bajas recompensas, en este caso sería de suma importancia detener la producción de anticuerpos con estas características. Otro posible caso puede ser que los anticuerpos produjeran recompensas medianamente buenas, en esta ocasión sería importante modificar parámetros como  $\sigma$ , y de esta manera aumentar la diversidad de anticuerpos, de tal manera

que este cambio pudiera producir una mayor cantidad de anticuerpos y dentro de todos ellos se encuentre alguno que produzca mayor recompensa.

## **2.2. Aprendizaje reforzado**

El aprendizaje reforzado, es una técnica que permite a una entidad aprender a partir de experiencias previas en el tiempo, este tipo de aprendizaje está estrechamente ligada con la programación dinámica y está constituida por conceptos importantes que permiten al ente aprender, estos conceptos son la política de comportamiento, que básicamente indica al ente que acciones debe tomar según el estado en el que se encuentre; es decir su comportamiento; la función de recompensa, que da una retroalimentación instantánea de la acción escogida y que está definida por el objetivo u objetivos a alcanzar.

Este método ha sido ampliamente utilizado en aplicaciones robóticas y sus resultados han sido favorables, probando así su competencia como método para que un ente pueda aprender ciertas habilidades. Incluso en algunas ocasiones ha sido posible demostrar que su desempeño es mucho mejor que el de código programado meramente a mano y de forma explícita sobre una tarea a realizar [15], ha sido utilizado en robots móviles, para permitir que los robots aprendan los aspectos más básicos de la tarea que están por realizar, en lugar de programar el mismo robot de manera explícita indicando paso por paso [16], también han sido exitosamente aplicados estos métodos a sistemas robóticos de visión [17] e incluso se ha logrado que lleven a cabo una tarea tan difícil como lo es la locomoción de un robot bípedo para que este camine [18]. Como se puede observar el aprendizaje reforzado no es una técnica nueva y está ya ha sido estudiada y aplicada de manera extensa en una amplia gama de escenarios y aplicaciones con resultados positivos, basados en [19] se hará un breve resumen de las características importantes de dicho método.

Como se puede leer en el texto anteriormente citado, el aprendizaje reforzado es seleccionacional, involucra probar y seleccionar entre un grupo de alternativas al comparar consecuencias y también es asociativa debido a que aquellas alternativas encontradas son asociadas con situaciones particulares. Es importante resaltar la unión de estos dos términos anteriores, dado que se combina la búsqueda y la memoria con el objetivo de buscar y seleccionar acciones acordes a cada situación y recordar cuales acciones funcionaron mejor y en qué momentos.

Adicionalmente el aprendizaje reforzado está también altamente involucrado con dos procesos importantes; la exploración y explotación. El primer término, exploración está ligado a la acción de probar nuevas acciones que no se han usado con anterioridad, con el objetivo de descubrir nuevas situaciones con posibles mejores recompensas. La explotación por otro lado se refiere al uso extensivo del conocimiento ya adquirido.

*Política de comportamiento:* La política de comportamiento define el comportamiento del agente, puede ser tan simple como una tabla de relaciones estado-acciones. O en

escenarios más complicados puede ser un proceso de búsqueda entre espacios de estados y espacios de acciones que involucran un alto procesamiento de datos. Este es un proceso naturalmente estocástico debido a que parte de las decisiones que se toman son a partir de procesos predecibles, pero al mismo tiempo otra parte de estos procesos son aleatorios.

*Función de recompensa:* La función de recompensa es específica para cada tarea o tareas a realizar. Esta función calcula que tan atractivo es el estado actual percibido asignando una recompensa numérica, este valor es calculado de manera inmediata dando así una retroalimentación de lo que es bueno o no para el agente.

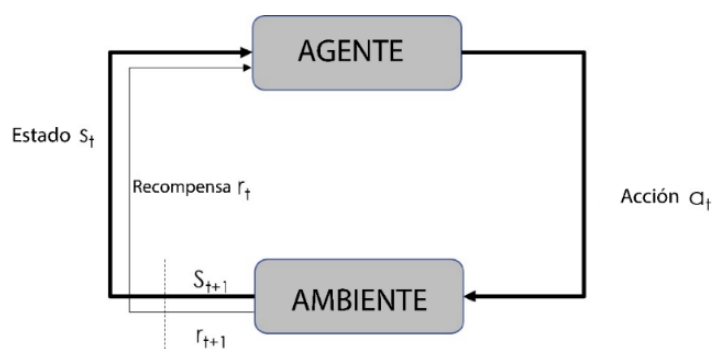


Fig. 4. Diagrama básico de los agentes en la dinámica de aprendizaje reforzado.

### 2.3. Aprendizaje por demostración

El aprendizaje por demostración es una técnica que permite enseñar a partir de ejemplos. Estos ejemplos pueden proveerse por una gran variedad de formas; a partir de los mismos sensores instalados en el robot [20], con la ayuda de un sistema externo con sensores similares o iguales a los instalados en el robot o ente [21], con sistemas de visión [22] entre otros [23]. Esta técnica es también conocida como “programación por demostración” o “programación por enseñanzas”, y en contraste con la programación clásica donde un experto codificador debe escribir un algoritmo lo suficientemente explícito sobre la tarea que se desea, en el aprendizaje por demostración casi cualquier humano puede auxiliar en la programación del ente que aprende en base a los ejemplos.

Es así como el robot asume el rol de estudiante y el humano el rol de experto o maestro en nuestro caso particular se usó un guante de datos que permite recabar información sobre la posición y fuerza ejercida por cada dedo por medio de sensores embebidos en el guante, permitiendo así recabar información a manera de ejemplos que será usada por el algoritmo CODA posteriormente. La siguiente sección ahonda en el sistema de pruebas que se diseñó y en su funcionamiento de tal manera que se pudieran obtener datos adecuados para ser usados por la aproximación en conjunto que se propone.



### **3. La plataforma de pruebas**

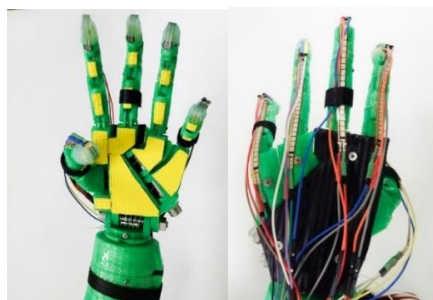
Con el objetivo de comprobar el algoritmo diseñado se escogió una de las tareas que más reto presenta en la robótica y esto es producir poses correctas para la manipulación de objetos. El sistema entero está constituido por un guante de datos y una mano robótica con forma antropomórfica, ambos con sensores de presión y de flexión que permiten obtener datos de posición y fuerza para cada dedo, adicionalmente en el sistema de la mano robótica se instalaron sensores de corriente que permitieran monitorear los actuadores para evitar sobre corrientes y daños.

#### **3.1. La mano antropomórfica**

El diseño de la mano fue tomado de un Proyecto internacional de código abierto llamado InMoov, desarrollado por el diseñador francés Gael Langevin. La mano en su totalidad está fabricada por una impresora 3D y como mecanismos que emulan los tendones utiliza hilo de poliamida para lograr el movimiento de los 5 dedos además de un sistema de engranes para un movimiento de 180° en la muñeca.

Como actuadores se instalaron servomotores de alto torque que jalan el hilo de poliamida con un diámetro de .90 mm.

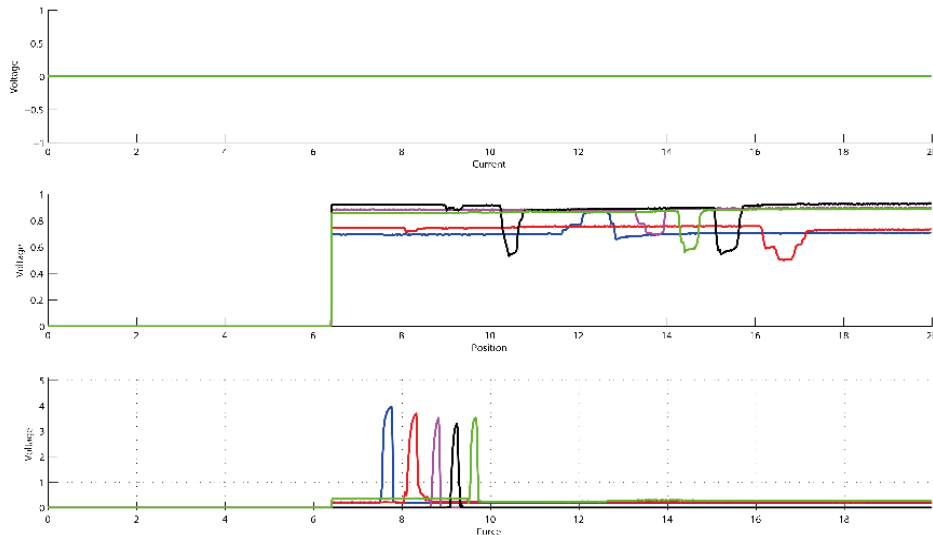
La mano que se usó para las pruebas cuenta con 16 grados de libertad y 6 grados actuados, haciendo que la configuración de dicho robot sea subactuado. La mano original, no está diseñada para ser usada con el tipo de sensores que se le instalaron, es por eso que se debieron hacer ciertas modificaciones para que los sensores pudieran estar embebidos en la mano y proporcionar datos útiles. La figura 5 muestra el resultado de las modificaciones y el sistema finalizado.



**Fig. 5.** Mano Robótica con sensores de presión y flexión instalados en el anverso y reverso.

El algoritmo no recibirá ninguna información paramétrica que tenga que ver con las dimensiones de la mano o los links que la conforman. La única información que recibirá es la obtenida por los sensores embebidos en el guante mientras es usado por un humano.

La figura 6 muestra un ejemplo de los datos obtenidos cuando los sensores de presión y flexión son excitados al ser presionados y flexionados respectivamente en el sistema de la mano antropomórfica, toda esta información puede ser recopilada por un microcontrolador y enviada a MATLAB donde se despliega de manera gráfica y casi instantánea.



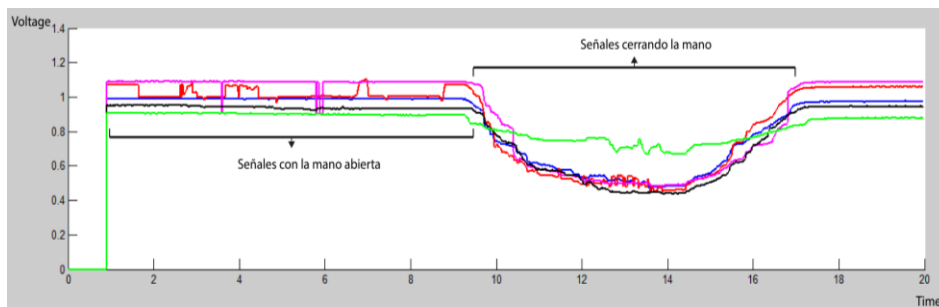
**Fig. 6.** De arriba a abajo; valores de los sensores de corriente cuando los motores están apagados, Sensores de posición cuando los dedos son movidos de su posición de reposo a una posición aleatoria y de regreso, sensores de fuerza cuando son presionados manualmente.

De la imagen superior se observa que los datos recolectados por el sistema están libres de ruido que afecte sus lecturas y que las lecturas de los sensores son coherentes, de tal manera que el sistema construido proporciona una plataforma de prueba lo suficientemente confiable.

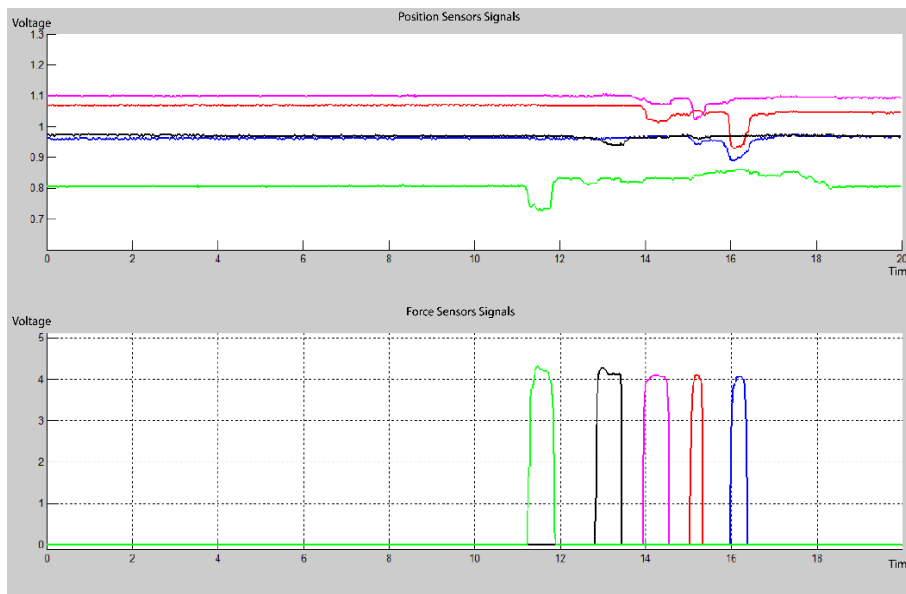
### 3.2. El guante de datos

El guante de datos es el instrumento que permite hacer uso del aprendizaje por demostración. Este sistema también cuenta con sensores de flexión para la posición y de presión para la fuerza de tal manera que pueda extraer esta información mínima requerida para reproducir un agarre. Se usó un guante genérico y un total de diez sensores, dos por cada dedo; enviando las lecturas de los sensores a MATLAB por medio de un microcontrolador. La figura 7 muestra las señales de los sensores de posición en una secuencia de dos movimientos, primero con la mano abierta y posteriormente con la mano cerrada, en la imagen se observa de manera clara las dos secuencias, este comportamiento

nos permite concluir que el sistema proporciona información clara y fácilmente reconocible e interpretable.



**Fig. 7.** Ejemplo de datos obtenidos durante un ejercicio de prueba al mantener abierta la mano y posteriormente cerrarla, usando el guante de datos y la interfaz en MATLAB.



**Fig. 8.** Gráficas obtenidas al hacer prueba de funcionalidad a cada sensor instalado en el guante de datos. La gráfica superior muestra la variación que sufren los valores al mover cada uno de los dedos de su posición de reposo. La gráfica inferior muestra el comportamiento de los valores al tocar una superficie con los sensores de presión de manera individual.

La figura 8 muestra otra prueba hecha a los sensores de fuerza instalados en el guante, las señales que se muestran son obtenidas al hacer que cada dedo alcance una superficie plana, probando así que cada sensor en cada dedo y es fácilmente perceptible cuando tiene

contacto con un objeto permitiendo entregar datos suficientemente consistentes al algoritmo para ser usados como antígenos.

#### 4. Pruebas y resultados

Para las pruebas con el sistema y el algoritmo se utilizó un set de figuras geométricas bien conocidas. La figura 9 muestra el conjunto entero de dichas figuras y algunos detalles que permiten tener una idea del tamaño en centímetros de sus dimensiones.

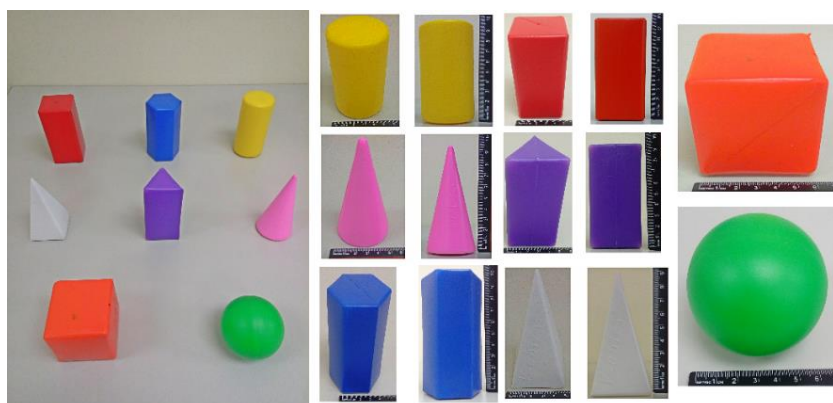


Fig. 9. Set de figuras geométricas para el uso en el protocolo de pruebas del algoritmo CODA.

El protocolo de pruebas es el siguiente;

1. Al seleccionarse el objeto se agarra por un experto humano que use el guante de datos. La información obtenida por los sensores funje como patógeno y es el ejemplo que se introduce al sistema.
2. Con los datos obtenidos se genera diversidad, usando el ejemplo y pasándolo por un proceso de clonación/mutación para obtener nuevos pares de datos.
3. Una vez generado un set de datos por medio del mecanismo emulado del SIN, se evalúan dichos datos por medio de las funciones de recompensa para eliminar datos que no cumplan con los parámetros deseados.
4. Aquellos datos mejor evaluados son utilizados por el sistema para producir el agarre de objetos.

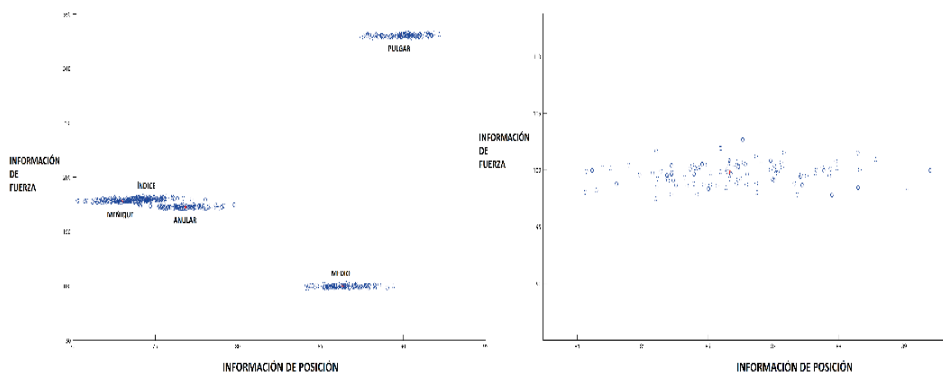
La tabla 2 resume la información obtenida por el guante de datos, como puede observarse, los valores de posición y fuerza ( $\theta$ ,  $\Sigma$ ) fueron recopilados por cada dedo, es importante resaltar que estos valores no están en unidades relacionadas a sus magnitudes físicas, son datos numéricos asignados por el convertidor análogo digital del microcontrolador y se manejaron de esta manera durante todo el proceso.

**Tabla2.** Ejemplos de agarres por cada figura geométrica obtenidos a partir del guante de datos.

$\theta$	$\Sigma$	$\theta$	$\Sigma$	$\theta$	$\Sigma$	$\theta$	$\Sigma$	DEDOS
74.00	23.67	84.33	0.00	87.00	127.00	71.83	306.17	<b>Meñique</b>
80.33	266.67	98.67	0.00	101.17	102.67	72.67	213.83	<b>Anular</b>
90.17	249.33	106.17	266.00	107.67	95.17	82.83	202.17	<b>Medio</b>
81.33	0.00	83.00	336.33	89.00	274.67	70.83	356.67	<b>Índice</b>
80.00	392.50	89.17	245.00	88.50	247.50	81.50	359.50	<b>Pulgar</b>
Esféra		Cubo		Prisma Rectangular		Cilindro		DEDOS
$\theta$	$\Sigma$	$\theta$	$\Sigma$	$\theta$	$\Sigma$	$\theta$	$\Sigma$	
73.00	178.33	66.83	0.00	76.17	0.00	74.50	0.00	<b>Meñique</b>
76.83	172.50	70.00	84.67	91.50	0.00	87.17	0.00	<b>Anular</b>
86.33	99.83	80.17	135.00	106.50	0.00	102.33	270.83	<b>Medio</b>
74.50	180.33	63.00	317.50	93.00	297.50	89.67	310.83	<b>Índice</b>
89.83	330.17	88.50	0.00	91.50	203.17	92.67	373.33	<b>Pulgar</b>
Prisma Hexagonal		Prisma Triangular		Pirámide		Cono		

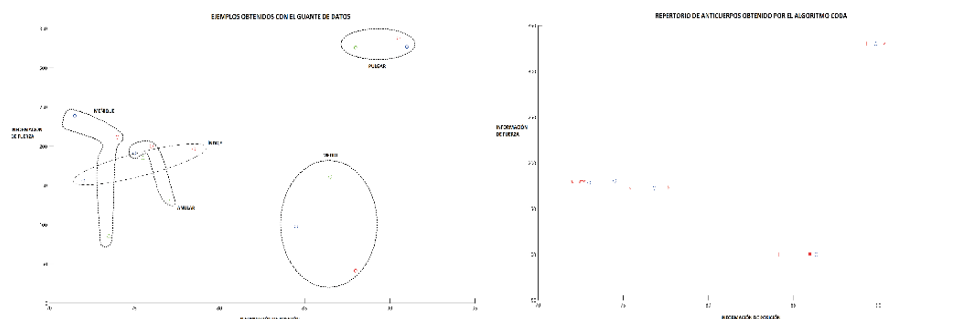
Estos datos se introdujeron al procedimiento de clonación/mutación del algoritmo CODA para producir N anticuerpos.

El resultado de introducir los datos al algoritmo con el ejemplo del prisma hexagonal produjo 500 nuevos y diferentes pares de datos, estos se muestran en la figura 10 en la imagen de la izquierda, el asterisco que se observa en cada uno de los conjuntos es el ejemplo que se le dio al algoritmo como información inicial, los círculos son los anticuerpos producidos. En la misma figura en la imagen de la derecha se observa como los anticuerpos tienen una distribución normal multivariable como se explicó en la sección uno.



**Fig. 10.** Izquierda, ejemplos (asteriscos) y anticuerpos (círculos) obtenidos por cada dedo a partir del procedimiento de generación de diversidad del algoritmo CODA. Derecha, vista detallada de anticuerpos producidos para el dedo medio.

Es importante indicar que los anticuerpos hasta este punto no han sido evaluados, solamente se ha generado diversidad, el siguiente paso es introducir los datos al proceso de selección negativa para evaluar su posible efectividad con la función de recompensa. La función de recompensa toma en consideración los valores de  $\theta$  y  $\Sigma$  para hacer la evaluación por cada par de datos de cada dedo, finalmente la recompensa obtenida por cada dedo es sumada hasta obtener por cada conjunto de 5 dedos un valor que oscila entre 0 y 5, siendo 5 el máximo valor posible y 0 el mínimo. Aquellos datos que cumplan con el criterio se guardan en el repertorio de anticuerpos, conteniendo así todos aquellos que se calificaron como más capaces. La figura 11 muestra todos aquellos anticuerpos que fueron calificados como aptos para el ejemplo del prisma hexagonal obteniendo un total de 6 posibles agarres diferentes de un total de 100 generados en un comienzo. En la misma imagen a la derecha se observan otros tres ejemplos capturados por el guante de datos que al ser comparados con los anticuerpos producidos por el algoritmo se observa que tienen distribuciones similares, sugiriendo que al estar en las mismas áreas que aquellos producidos por humanos reales, pertenecen a agarres correctos para el objeto entrenado. Este tipo de clusterización se desea embeber en el algoritmo para posteriormente poder obtener información relevante a la forma del objeto sin necesidad de una retroalimentación visual. Finalmente, para corroborar dicha información se usaron estos datos con la mano antropomórfica y al enviarlos de manera individual con la figura que se usó en el entrenamiento, se obtuvieron resultados positivos con un 83% de efectividad; es decir 5 de las seis posturas acertaron en agarrar la pieza sin que esta se cayera fácilmente o se moviera de manera irregular.



**Fig. 11.** Izquierda, tres ejemplos obtenidos a través de humanos usando el guante de datos. Derecha, ejemplos que se le dieron al sistema y repertorio que produjo con los anticuerpos más aptos.

## 5. Conclusiones

El algoritmo que se presentó en este documento, fue diseñado para emular al Sistema natural inmune en ciertos aspectos computacionalmente importantes para resolución de

problemas y su implementación en aplicaciones robóticas, como es el caso de la manipulación o agarre de objetos. Con el objetivo de hacer que el algoritmo tuviera un desempeño superior se complementaron los procedimientos emulados con el uso de métodos como el aprendizaje reforzado y el aprendizaje por medio de ejemplos de tal manera que pudieran ser complementarios con los sistemas inmunes artificiales. El algoritmo fue capaz de dar un porcentaje de aciertos de 83% de casos positivos con un prisma hexagonal y un 100% de casos positivos con una prueba posterior con un cubo. Adicionalmente se observa que los datos tienden a agruparse según la forma, sugiriendo así que al hacer uso de un algoritmo de clusterización y de un clasificador. Finalmente, el sistema demostró ser lo suficientemente capaz y confiable de ser usado para probar el algoritmo, aportando datos limpios y coherentes que permitieron implementar este algoritmo en la mano robótica de forma antropomórfica.

## **Bibliografía**

1. Yang, X. S.: *Swarm intelligence and bio-inspired computation: theory and applications*. Newnes (2013)
2. Boyan, J., Freitag, D., Joachims, T.: A machine learning architecture for optimizing web search engines. In: *AAAI Workshop on Internet Based Information Systems* (1996)
3. Agichtein, E., Brill, E., Dumais, S.: Improving web search ranking by incorporating user behavior information. In: *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM (2006)
4. Jiang, L., Liu, D. Y., Yang, B.: Smart home research. In: *Proceedings of the Third Conference on Machine Learning and Cybernetics SHANGHAI* (2004)
5. Tapia, E.M., Intille, S.S., Larson, K.: *Activity recognition in the home using simple and ubiquitous sensors*. Springer (2004)
6. Tan, C. P.: Classification of multi-temporal SAR images for rice crops using combined entropy decomposition and support vector machine technique. *Progress in Electromagnetics Research*, Vol. 71, pp. 19–39 (2007)
7. Kurata, K., Eguchi, N.: Machine learning of fuzzy rules for crop management in protected cultivation. *Transactions of the ASAE*, Vol. 33, No. 4, pp. 1360–1368 (1990)
8. Farmer, J. D., Packard, N. H., Perelson, A. S.: The immune system, adaptation, and machine learning. *Physica D: Nonlinear Phenomena*, Vol. 22, No. 1, pp. 187–204 (1986)
9. Forrest, S.: Self-nonsel self discrimination in a computer. *IEEE* (1994)
10. Cutello, V.: Clonal selection algorithms: a comparative case study using effective mutation potentials. In *Artificial Immune Systems*, Springer, pp. 13–28 (2005)

11. De Castro, L. N., Timmis, J.: Artificial immune systems: a new computational intelligence approach. Springer Science & Business Media (2002)
12. DasGupta, D.: An overview of artificial immune systems and their applications. Springer (1993)
13. Abbas, A. K., Lichtman, A. H., Pillai, S.: Basic immunology: functions and disorders of the immune system. Elsevier Health Sciences (2012)
14. Rebollo, D. R. R., Cruz, P. P., Molina, A.: CODA Algorithm: An Immune Algorithm for Reinforcement Learning Tasks. In: Recent Advances in Robotic Systems, InTech (2016)
15. Mahadevan, S., Connell, J.: Automatic programming of behavior-based robots using reinforcement learning. Artificial intelligence, Vol. 55, No. 2-3, pp. 311–365 (1992)
16. Mataric, M. J.: Reinforcement learning in the multi-robot domain. In: Robot colonies, Springer, pp. 73–83 (1997)
17. Asada, M.: Purposive behavior acquisition for a real robot by vision-based reinforcement learning. In: Recent Advances in Robot Learning, Springer, pp. 163–187 (1996)
18. Benbrahim, H., Franklin, J. A.: Biped dynamic walking using reinforcement learning. Robotics and Autonomous Systems, Vol. 22. No. 3, pp. 283–302 (1997)
19. Sutton, R. S., Barto, A. G.: Reinforcement learning: An introduction. MIT press (1998)
20. Pastor, P.: Learning and generalization of motor skills by learning from demonstration. In: Robotics and Automation, ICRA'09, IEEE International Conference on, IEEE (2009)
21. Zöllner, R., Rogalla, O., Dillmann, R.: Integration of tactile sensors in a programming by demonstration system. In: Robotics and Automation, proceedings 2001 ICRA, IEEE International Conference on, IEEE (2001)
22. Atkeson, C. G., Schaal, S.: Learning tasks from a single demonstration. In: Robotics and Automation, proceedings, IEEE International Conference on, IEEE (1997)
23. Ijspeert, A. J., Nakanishi, J., Schaal, S.: Learning rhythmic movements by demonstration using nonlinear oscillators. In: Proceedings of the IEEE/rsj int. conference on intelligent robots and systems (2002)