



Sistemas Inmunes Artificiales (AIS)

Artificial Immune Systems

Prof. D.Sc. BARSEKH-ONJI Aboud

Facultad de Ingeniería
Universidad Anáhuac México

12 de noviembre de 2025



Agenda

- 1 Fundamentos Biológicos del Sistema Inmune Natural (NIS)
 - Arquitectura de la Defensa: Innata vs. Adaptativa
 - Actores Celulares Clave: El Mapeo para la Computación
 - Principios Fundamentales para la Abstracción Computacional
- 2 Paradigmas de los Sistemas Inmunes Artificiales (AIS)
- 3 Algoritmo de Selección Negativa (NSA) para Detección de Anomalías
- 4 Algoritmo de Selección Clonal (CLONALG) para Optimización



De la Biología a la Computación

La Inspiración: El Sistema Inmune Natural (NIS)

Para comprender los Sistemas Inmunes Artificiales (AIS), primero debemos analizar el sistema biológico que les sirve de inspiración: el **Sistema Inmune Natural (NIS)**.



De la Biología a la Computación

¿Qué es el NIS?

Es un sistema de defensa de una complejidad asombrosa, caracterizado por su capacidad de:

- Aprendizaje
- Memoria
- Reconocimiento de patrones
- Descentralización



De la Biología a la Computación

La Metáfora Computacional

Los algoritmos de AIS son, en esencia, metáforas computacionales de los procesos biológicos del NIS.



Arquitectura de Defensa de Dos Velocidades

Inmunidad Innata (Rápida)

- **Primera línea de defensa.**
- Es una respuesta **rápida** y **no específica**.
- Está preconfigurada para reaccionar a patrones comunes de patógenos.
- Su función es contener la infección y *'proporcionar el tiempo necesario para activar el sistema adaptativo'*.



Arquitectura de Defensa de Dos Velocidades

Inmunidad Adquirida (Adaptativa)

- Es la respuesta **especializada**.
- Es más lenta de activar, pero genera una defensa **a medida** y con **memoria**.
- **Humoral (Células B)**: Secretan Anticuerpos para neutralizar invasores *fuera* de las células.
- **Celular (Células T)**: Identifican y destruyen las *propias células* del cuerpo que han sido infectadas.

Arquitectura de Defensa de Dos Velocidades

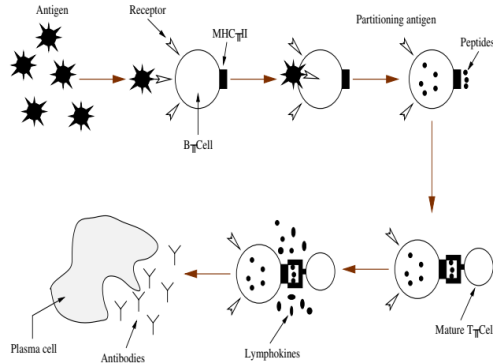


Figura 1: Diferentes etapas de la respuesta inmune.



Arquitectura de Defensa de Dos Velocidades

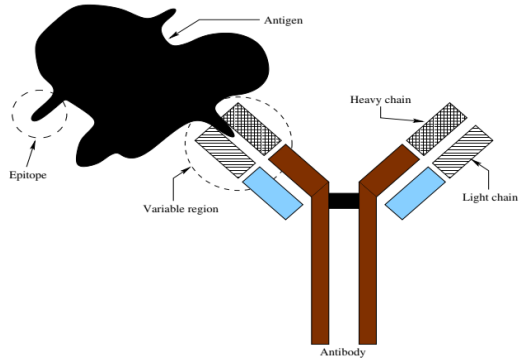


Figura 2: LA respuesta de los anticuerpos.



Abstracción: Defensa de 'Dos Velocidades'

La Implicación Computacional

Esta arquitectura biológica de 'dos velocidades' sugiere un modelo computacional óptimo para problemas complejos, como la ciberseguridad. Los AIS no tienen por qué ser monolíticos.



Abstracción: Defensa de 'Dos Velocidades'

Modelo Híbrido

Un diseño computacional fiel a la biología podría ser un sistema híbrido:

- 1 Un algoritmo de detección de anomalías de **bajo coste** (inspirado en la **inmunidad innata**) maneja la primera línea.
- 2 Al detectar una amenaza, **activa** un algoritmo más robusto y costoso (inspirado en la **inmunidad adaptativa**).
- 3 Este segundo algoritmo caracteriza la amenaza y genera una respuesta específica con memoria.



Actores Clave (1/2): Antígenos y Células B

Antígenos (Ag)

- **Bio:** Las moléculas 'invasoras' (generalmente proteínas) que el sistema reconoce como 'no-propias'.
- **IC:** Son los **datos a clasificar**, optimizar o reconocer (ej. un patrón de tráfico de red anómalo, un virus informático).

Linfocitos B (Células B)

- **Bio:** Producen **Anticuerpos (Ab)**, que son las moléculas de reconocimiento de patrones por excelencia.
- **IC:** La interacción Antígeno-Anticuerpo es la inspiración directa del **Algoritmo de Selección Clonal (CLONALG)**.



Actores Clave (2/2): Células T y Células Dendríticas

Linfocitos T (Células T)

- **Bio:** Maduran en el Timo. Durante su maduración, aprenden a **no** atacar a las células 'propias' del cuerpo. Este proceso se llama **auto-tolerancia**.
- **IC:** Este mecanismo es la inspiración directa del **Algoritmo de Selección Negativa (NSA)**.



Actores Clave (2/2): Células T y Células Dendríticas

Células Dendríticas (CD)

- **Bio:** Son las 'centinelas' y el puente entre la inmunidad innata y la adaptativa.
- **Bio:** Detectan un patógeno, lo procesan y lo 'presentan' a las Células T, activando la respuesta adaptativa.
- **IC:** Esta función de fusión de señales y activación contextual es la inspiración del **Algoritmo de Células Dendríticas (DCA)**.



Principios Computacionales (1/4)

Principio 1: Discriminación Propio/No-Propio

- **Biología:** El sistema debe distinguir entre componentes del cuerpo ('propio' o *self*) e invasores ('no-propio' o *non-self*).
- **Abstracción (IC):** Es el problema de la **detección de anomalías** (one-class classification).



Principios Computacionales (2/4)

Principio 2: Selección Clonal y Maduración de la Afinidad

- **Biología:** Cuando un Linfocito B reconoce un antígeno, prolifera (clones). Estos clones sufren una hipermutación. Los clones que 'mejoran' su afinidad (capacidad de enlace) son seleccionados para sobrevivir. Es un proceso darwiniano.
- **Abstracción (IC):** Es la base de los algoritmos de **optimización** de AIS.



Principios Computacionales (3/4)

Principio 3: Memoria Inmunológica

- **Biología:** Tras una infección, células de alta afinidad se convierten en células de memoria de larga duración, permitiendo una respuesta futura 'rápida, enérgica y específica'.
- **Abstracción (IC):** Es el mecanismo de **aprendizaje** y adaptación del sistema.



Principios Computacionales (4/4)

Principio 4: Teorías Alternativas (Red y Peligro)

- **Teoría de la Red Idiopática:** El sistema inmune se reconoce a sí mismo, creando una red interna de regulación y supresión.
- *Inspiración IC:* **Redes Inmunes Artificiales (aiNet).**



Principios Computacionales (4/4)

Principio 4: Teorías Alternativas (Red y Peligro)

- **Teoría de la Red Idiopática:** El sistema inmune se reconoce a sí mismo, creando una red interna de regulación y supresión.
- *Inspiración IC:* **Redes Inmunes Artificiales (aiNet).**
- **Teoría del Peligro:** El sistema inmune no responde a lo 'no-propio', sino al '**peligro**'. Las Células Dendríticas actúan como 'fusionadoras de datos', combinando la detección de un antígeno con señales de estrés celular.
- *Inspiración IC:* **Algoritmo de Células Dendríticas (DCA).**



Abstracción: Del 'Propio' al 'Contexto'

La Evolución de la Abstracción

El modelo simplista 'Propio/No-Propio' (Principio 1) es frágil. Computacionalmente, un algoritmo simple de Selección Negativa (NSA) basado en una definición estática de 'propio' fracasa en entornos dinámicos (como un sistema informático), generando una avalancha de falsos positivos.



Abstracción: Del 'Propio' al 'Contexto'

La Solución del DCA: Cambiar la Pregunta

El Algoritmo de Células Dendríticas (DCA) resuelve esto cambiando la pregunta fundamental:

- **Pregunta del NSA:** *¿Es esto 'propio'?* (Coincidencia de patrones).



Abstracción: Del 'Propio' al 'Contexto'

La Solución del DCA: Cambiar la Pregunta

El Algoritmo de Células Dendríticas (DCA) resuelve esto cambiando la pregunta fundamental:

- **Pregunta del NSA:** *¿Es esto 'propio'?* (Coincidencia de patrones).
- **Pregunta del DCA:** *¿Está este patrón ocurriendo en un **contexto** de 'peligro'?* (Fusión y correlación de datos).



Abstracción: Del 'Propio' al 'Contexto'

El Paradigma Moderno

Por esto, el DCA se centra en la 'fusión y correlación de datos' de múltiples señales de entrada, un paradigma mucho más robusto para la detección de anomalías en el mundo real.



Agenda

- 1 Fundamentos Biológicos del Sistema Inmune Natural (NIS)
 - Arquitectura de la Defensa: Innata vs. Adaptativa
 - Actores Celulares Clave: El Mapeo para la Computación
 - Principios Fundamentales para la Abstracción Computacional
- 2 Paradigmas de los Sistemas Inmunes Artificiales (AIS)
- 3 Algoritmo de Selección Negativa (NSA) para Detección de Anomalías
- 4 Algoritmo de Selección Clonal (CLONALG) para Optimización



De la Biología a la Computación

El Proceso de Abstracción

Los Sistemas Inmunes Artificiales (AIS) son una familia de algoritmos dentro de la Inteligencia Computacional (IC).

- A diferencia de otros métodos de IC, los AIS se centran en metáforas de:
 - Evolución Darwiniana (selección clonal)
 - Aprendizaje (memoria)
 - Reconocimiento de Patrones (afinidad antígeno-anticuerpo)



De la Biología a la Computación

Los 4 Paradigmas Principales

No existe un único 'algoritmo AIS', sino cuatro familias o paradigmas principales, cada uno inspirado en un principio biológico diferente:

- 1 Algoritmos de Selección Negativa (NSA)
- 2 Algoritmos de Selección Clonal (CSA / CLONALG)
- 3 Redes Inmunes Artificiales (INA / aiNet)
- 4 Algoritmos de Células Dendríticas (DTA / DCA)



Mapeo: Biología → Computación (La 'Piedra Roseta')

Conectando Conceptos

La siguiente tabla actúa como una 'piedra roseta', conectando los conceptos biológicos de la Sección 1 con los paradigmas algorítmicos que estudiaremos.

Mapeo: Biología → Computación (La 'Piedra Roseta')

Principio Biológico	Proceso/Célula Clave	Abstracción Computacional (El Problema)	Algoritmo AIS
Auto-Tolerancia	Maduración de Células T en el Timo	Detección de Anomalías (One-Class): Definir datos 'propios' y generar detectores para todo lo 'no-propio'.	NSA (Algoritmo de Selección Negativa)
Maduración de la Afinidad	Selección y proliferación de Células B	Optimización de Funciones: Usar la afinidad (fitness) para clonar (explotar) y mutar (explorar) soluciones candidatas.	CLONALG (Algoritmo de Selección Clonal)

Mapeo: Biología → Computación (La 'Piedra Roseta')

Red Idiopática	Interacción y supresión mutua de Anticuerpos	Clustering y Compresión de Datos: Crear una red de 'prototipos' (anticuerpos) que se auto-suprimen para cubrir el espacio de datos.	aiNet (Red Inmune Artificial)
Teoría del Peligro	Fusión de señales por Células Dendríticas	Detección de Anomalías Contextual: Correlacionar múltiples flujos de datos para determinar si un evento es anómalo <i>dado su contexto</i> .	DCA (Algoritmo de Células Dendríticas)



Agenda

- 1 Fundamentos Biológicos del Sistema Inmune Natural (NIS)
 - Arquitectura de la Defensa: Innata vs. Adaptativa
 - Actores Celulares Clave: El Mapeo para la Computación
 - Principios Fundamentales para la Abstracción Computacional
- 2 Paradigmas de los Sistemas Inmunes Artificiales (AIS)
- 3 Algoritmo de Selección Negativa (NSA) para Detección de Anomalías
- 4 Algoritmo de Selección Clonal (CLONALG) para Optimización



Inspiración y Objetivo 1/3

Definición

El Algoritmo de Selección Negativa (NSA) es uno de los paradigmas fundacionales de los AIS, diseñado específicamente para la **detección de anomalías** (one-class classification).

Inspiración y Objetivo 2/3

Inspiración Biológica: El Timo

El NSA está inspirado en el proceso de maduración de las **Células T** en el timo:

- 1 Se generan 'detectores' (Células T) de forma aleatoria.
- 2 Estos detectores se prueban contra las células 'propias' (*self*) del cuerpo.
- 3 Si un detector reacciona (coincide) con una célula 'propia', es eliminado (selección negativa).
- 4 Solo los detectores que **no** reaccionan contra lo 'propio' son liberados al cuerpo para monitorear invasores 'no-propios' (*non-self*).



Inspiración y Objetivo 3/3

Objetivo Computacional

El objetivo es idéntico: crear un conjunto de 'detectores' que puedan discriminar entre 'propio' (comportamiento normal del sistema) y 'no-propio' (comportamiento anómalo).



El Algoritmo Formal (Pseudocódigo Canónico) 1/2

Fase 1: Censura (Entrenamiento): Genera el conjunto de detectores.

Algorithm 1 Generar Detectores

```
1: Definir S (conjunto de patrones 'propios').  
2: Generar D (conjunto vacío de detectores).  
3: while  $|D| < n\_detectores\_deseados$  do  
4:    $d \leftarrow \text{generar\_detector\_aleatorio}()$   
5:   if  $d$  no coincide con ningún  $s$  en S then  
6:     Añadir  $d$  a D  
7:   end if  
8: end while  
9: return D
```

El Algoritmo Formal (Pseudocódigo Canónico) 2/2

Fase 2: Monitorización (Clasificación): Usa los detectores para clasificar nuevos datos.

Algorithm 2 Monitorear Datos

```
1: Recibir D (de la Fase 1).
2: for all nuevo patrón m a monitorear do
3:   anomalía ← False
4:   for all detector d en D do
5:     if m coincide con d then
6:       anomalía ← True
7:       break
8:     end if
9:   end for
10:  if anomalía then
11:    return 'Anomalía (No-Propio)'
12:  else
13:    return 'Normal (Propio)'
14:  end if
15: end for
```



Algoritmo de Selección Negativa (NSA): Ejemplo de aplicación

Aplicación en MATLAB

Archivo adjunto a la presentación de esta clase.



Agenda

- 1 Fundamentos Biológicos del Sistema Inmune Natural (NIS)
 - Arquitectura de la Defensa: Innata vs. Adaptativa
 - Actores Celulares Clave: El Mapeo para la Computación
 - Principios Fundamentales para la Abstracción Computacional
- 2 Paradigmas de los Sistemas Inmunes Artificiales (AIS)
- 3 Algoritmo de Selección Negativa (NSA) para Detección de Anomalías
- 4 Algoritmo de Selección Clonal (CLONALG) para Optimización

Inspiración y Objetivo 1/2

Inspiración Biológica: Maduración de la Afinidad

La inspiración es el **Principio de Selección Clonal** y la **Maduración de la Afinidad** de las Células B.

- Cuando una Célula B se une a un antígeno, es **seleccionada**.
- La célula seleccionada prolifera (se **clona**) y sus clones sufren **mutaciones**.
- Los clones que, por azar, mejoran su afinidad (capacidad de enlace) son seleccionados para sobrevivir. Es un proceso de optimización darwiniano.



Inspiración y Objetivo 2/2

La Metáfora Computacional

El objetivo es encontrar soluciones óptimas a un problema (usualmente optimización multimodal).

- **Antígeno:** El problema a resolver (o la función de fitness).
- **Anticuerpo:** Una solución candidata al problema (ej. un vector de parámetros).
- **Afinidad:** El *fitness* de la solución (qué tan buena es).



El Algoritmo Formal (CLONALG)

Algoritmo de Aprendizaje CLONALG

El algoritmo combina la selección darwiniana con una tasa de mutación dinámica.

El Algoritmo Formal (CLONALG)

Algorithm 3 Pseudocódigo de CLONALG

- 1: Inicializar una población **Ab** de anticuerpos (soluciones) aleatorios.
- 2: **while** no se cumpla la condición de parada **do**
- 3: Presentar Antígeno (Problema).
- 4: **Evaluar Afinidad (Fitness)** de todos los anticuerpos en **Ab**.
- 5: **Seleccionar** los n anticuerpos con mayor afinidad.
- 6: **Clonar** los n mejores anticuerpos.
 - N° de Clones \propto Afinidad (los mejores se clonan más).
- 7: **Hipermutar** los clones.
 - Tasa de Mutación $\propto \frac{1}{\text{Afinidad}}$ (los mejores mutan menos).
- 8: **Re-seleccionar** los mejores clones y guardarlos como 'memoria' si superan a sus padres.
- 9: **Reemplazar** los d peores anticuerpos por nuevas soluciones aleatorias (para mantener la diversidad).
- 10: **end while**
- 11: **return** El mejor anticuerpo (solución) encontrado.



El Núcleo de CLONALG: Exploración vs. Explotación 1/3

El Balance Dinámico

El núcleo intelectual de CLONALG reside en la interacción de los pasos de clonación y mutación, que balancean la búsqueda de soluciones nuevas vs. el perfeccionamiento de las existentes.



El Núcleo de CLONALG: Exploración vs. Explotación 2/3

Explotación (Afinar Soluciones)

Soluciones de **alta afinidad** (buenas):

- → **Mucha** Clonación.
- → **Poca** Mutación.

Resultado: Permite un ajuste fino (como *hill climbing*) en la vecindad de una buena solución.

Exploración (Buscar Soluciones)

Soluciones de **baja afinidad** (malas):

- → **Poca** Clonación.
- → **Mucha** Mutación.

Resultado: 'Lanza' la solución a regiones completamente nuevas del espacio de búsqueda para encontrar otros óptimos locales.



El Núcleo de CLONALG: Exploración vs. Explotación 3/3

Diferencia con Algoritmos Genéticos

La principal diferencia es que la tasa de mutación es **dinámica** y depende del fitness de la solución, a diferencia de la tasa de mutación usualmente estática en un AG simple.



Algoritmo CLONALG: Ejemplo de aplicación

Aplicación en MATLAB

<https://www.mathworks.com/matlabcentral/fileexchange/67010-clonalg-clonal-selection-algorithm-for-optimization-problems>