

# Aprendizaje adaptativo en la tragedia de los comunes

Julián Andrés García\* y Luis Fernando Niño V.\*\*

## Adaptive Learning in the Tragedy of the Commons

### RESUMEN

La utilización conjunta de un recurso común, generalmente causa su extinción. Esto se conoce como la *tragedia de los comunes*. En este trabajo se analizan los efectos del aprendizaje adaptativo, usando programación genética, en esta clase de situaciones. En el marco de la teoría de juegos, aquí se consideró la interacción estratégica entre los jugadores y la dinámica de un ambiente vinculado fuertemente al juego. Se formuló un modelo de simulación para la situación de los *comunes*, con base en los resultados de un juego analítico. Posteriormente, se realizó una serie de experimentos computacionales para obtener reglas de juego evolucionadas, las cuales fueron analizadas en relación con los resultados predichos por el modelo analítico. Los resultados obtenidos fueron similares a aquellos predichos por la teoría de juegos clásica, pero no siempre condujeron a una tragedia.

### PALABRAS CLAVE

Programación genética, tragedia de los comunes, teoría de juegos

### ABSTRACT

The joint utilisation of a commonly owned resource often causes the resource to be overused, this is known as *The tragedy of the Commons*. This paper analyses the effects of adaptive learning in such kind of situations using genetic programming. In a game theoretical approach, the situation considers not only the strategic interaction among players, but also the dynamics of a changing environment linked strongly to the players' actions and payoffs. The results of an analytical game are used to formulate a simulation game for *the commons*, then a series of computational experiments are conducted, obtaining evolved game strategies that are examined in comparison with those predicted by the analytical model. The obtained results are similar to those predicted by classic game theory, but not always leading to a tragedy.

### KEY WORDS

Genetic Programming, tragedy of the commons, game theory

\* Departamento de Ingeniería de Sistemas e Industrial, Universidad Nacional de Colombia, Bogotá. julian.garcia@ieee.org

\*\* Departamento de Ingeniería de Sistemas e Industrial, Universidad Nacional de Colombia, Bogotá. lfninov@unal.edu.co

## 1. INTRODUCCIÓN

La utilización conjunta de un recurso de propiedad común causa con frecuencia la sobreexplotación del recurso. Esta situación se conoce como la *Tragedia de los comunes*. Este artículo analiza los efectos del aprendizaje adaptativo en tal tipo de situaciones, modelando y simulando agentes adaptativos mediante programación genética (PG).

La tragedia de los comunes ha sido ampliamente estudiada por economistas, sociólogos y, en general, teóricos de sistemas. La situación representa "una abstracción de referencia común para situaciones muy comunes en la vida real" [2], por ejemplo, la pesca en aguas internacionales, la polución, el calentamiento global, entre otras.

Los aspectos críticos de una situación de *comunes* son:

- Un recurso explotado por un grupo de individuos.
- Todos los individuos tienen libre acceso al recurso.
- Cuanto más intenso sea el uso del recurso en el presente, más limitado será el recurso en el futuro.

La última característica implica un compromiso (*tradeoff*) entre el consumo real en el presente y el posible consumo en el futuro. La cantidad venidera de recurso disponible dependerá del recurso disponible en el presente. Así, un individuo estaría dispuesto a sacrificar su consumo presente para obtener ganancias mayores en el futuro, pero este comportamiento "moderado" sólo es eficiente y efectivo en tanto todos los individuos asuman la misma posición. Cuando se consideran agentes que actúan procurando tan sólo su propio beneficio (agentes completamente racionales), la teoría de juegos ha mostrado que los recursos de propiedad común siempre serán explotados más allá del punto más deseable desde la óptica colectiva [3].

Diferentes técnicas de inteligencia artificial han mostrado ser útiles para analizar las relaciones existentes entre el aprendizaje y la cooperación en juegos. En particular, se han empleado algoritmos evolutivos para tratar de arrojar luz sobre los aspectos que condicionan emergencia del comportamiento cooperativo. La literatura reporta resultados en problemas clásicos como el dilema del prisionero iterado (DPI), que ha sido objeto de amplio estudio por los científicos de la compu-

tación. Se han explorado numerosos enfoques, incluidas versiones del juego con múltiples jugadores y conjuntos extendidos de estrategias. Todos estos esfuerzos apuntan a estudiar situaciones más realistas [6] [7].

El marco de los comunes es muy similar al del DPI porque una elección que puede ser percibida como la mejor opción para un individuo implica resultados pobres para la sociedad en pleno. Sin embargo, los comunes considera además un ambiente cambiante fuertemente vinculado al juego. Así, la situación es inherentemente dinámica, incluso más allá de los cambios en las acciones de los jugadores.

Las simulaciones computacionales de explotación de recursos comunes se han llevado a cabo usando técnicas distintas de la computación evolutiva. Se puede encontrar un ejemplo en [8], que utiliza agentes *swarm* para simular una tragedia de los comunes usando reglas de tipo *si-entonces*, y sugiere el uso de técnicas evolutivas para investigación posterior. Este trabajo pretende ser una contribución en la misma dirección. Las situaciones de la vida real involucran condiciones ambientales que tienden a ser altamente dinámicas. En este sentido, la situación de los comunes puede representar un marco realista e interesante. La pregunta por plantear es: ¿resuelve el comportamiento adaptativo la tragedia de los comunes?

El resto de este artículo está organizado como sigue: La Sección 2 introduce algunos resultados de la teoría de juegos en los que este trabajo se basa; particularmente, se presenta un juego dinámico<sup>1</sup> de  $n$  jugadores. La Sección 3 discute el modelo de simulación propuesto, y se describen los experimentos realizados basados en PG. La Sección 4 analiza los resultados experimentales. Las estrategias obtenidas se estudian en relación con la función de *fitness* y se comparan con las predicciones obtenidas por la teoría de juegos clásica. Finalmente, se enuncian algunas conclusiones en la Sección 5.

1. Un juego que considera no sólo la interacción estratégica entre los jugadores, sino también las interacciones entre los jugadores y el ambiente.

## 2. PRELIMINARES

En esta sección se presenta la versión de  $n$  jugadores de un juego, introducido por Mirman y Levhari [9]. El modelo es un juego dinámico que captura las características esenciales de un escenario de comunes. El argumento usado para explicar la tragedia es que "el equilibrio de Nash en el juego siempre llevará a un resultado que es peor que el más deseable socialmente" [3]. La versión que se presenta a continuación está basada fuertemente en la explicación dada en [10] para el caso de dos jugadores.

### 2.1 Marco matemático

Considere  $n$  jugadores que tienen libre acceso a un recurso común durante un período de tiempo particular. Sea  $Y_t$  la cantidad de recurso disponible en un tiempo  $t$ , y  $c_{it}$  la cantidad de recurso consumida por un jugador  $i$  en el tiempo  $t$ .

Claramente, para todos los valores de  $t$ , se tiene

$$0 \leq \sum_{i=1}^n c_{it} \leq y_t$$

Sea  $x_t$  el recurso no consumido en un tiempo  $t$ , luego

$$x_t = y_t - \sum_{i=1}^n c_{it}$$

Un aspecto importante en este modelo es que el recurso debe ser renovable, por tanto, una ley natural que actúa sobre el recurso debe tenerse en cuenta, introduciendo una constante  $\alpha$  en  $[0, 1]$ , que representa la tasa de renovación, de manera que

$$y_{t+1} = x_t^\alpha$$

El comportamiento natural del recurso, sin ninguna interferencia externa, está dominado por dos puntos fijos, 0 y 1 (ver [9] para detalles), donde 0 representa la total escasez del recurso y 1 representa un estado estable de sostenibilidad. Además, cuando  $x_t$  está en  $(0, 1)$ , valores mayores de  $x_t$  producen valores mayores de  $y_{t+1}$ ; por tanto la cantidad de recurso no consumido puede considerarse como una inversión.

El valor de juego (*payoff*) recibido por el jugador  $i$  en el tiempo  $t$  es dado por la función de utilidad

$$u_i^t = \log c_{it}$$

Dentro de este marco, la pregunta por plantear es: ¿implica la interacción estratégica entre los jugadores el sobreconsumo?

### 2.2 Óptimo social

¿Cuál es la política de consumo que permite a los jugadores en su conjunto ser tan felices como sea posible? El criterio de optimalidad social será la suma de los valores de juegos obtenidos por todos los jugadores. Así, el objetivo consiste en optimizar esta cantidad a lo largo del tiempo, para lo cual se usa inducción hacia atrás (*backward induction*) (ver detalles en [9]). Comenzando en el último período del juego, el consumo óptimo (desde el punto de vista social) se calcula retrospectivamente en cada período. Como los valores de juego se optimizan a través del tiempo, es necesario considerar un factor de descuento  $\delta = 1/(1+r)$ , donde  $r$  es la cantidad de una unidad de consumo que un individuo está dispuesto a ceder para poder consumir en el período presente, en lugar de hacerlo en el siguiente. Puede verificarse que  $\delta$  está en  $[0, 1]$ .

Procediendo de esta manera, al expresar los consumos como fracciones del recurso disponible  $y$ , es posible observar una expresión resultante para las fracciones de consumo, como se muestra en la Tabla 1.

Tabla 1  
Política de consumo correspondiente al óptimo social

Número de períodos	Fracción de consumo
1	$1/n$
2	$1/n (1 + \alpha\delta)$
3	$1/n (1 + \alpha\delta) + (\alpha\delta)^2$
.	.
.	.
.	.
T	$1/n (\sum_{i=0}^{T-1} (\alpha\delta)^i)$

Así, el consumo óptimo social es una función del tiempo restante por jugar  $T$ , y las variables  $n$ ,  $\delta$  y  $\alpha$ . Esta

regla determina qué fracción del recurso disponible debe consumir un jugador para optimizar la asignación del recurso desde el punto de vista social, como sigue:

$$f^s(T, \alpha, \delta, n) = \frac{1}{n \sum_{i=0}^{T-1} (\alpha\delta)^i}$$

Política de óptimo social

### 2.3 Equilibrio de Nash

Otra pregunta de interés es: cuando se asumen jugadores completamente racionales, ¿tiene lugar la política de óptimo social? Para afrontar esta pregunta, se encontrará la solución de Nash para el juego.

El cálculo del equilibrio de Nash requiere un supuesto adicional, el supuesto de sobreconsumo.

*Supuesto de sobreconsumo.* Cuando el consumo deseado de los jugadores es mayor que la cantidad disponible, el recurso se divide igualmente entre los jugadores.

Para cada jugador se calcula la función de mejor respuesta (una que indica a un jugador qué hacer, dadas las acciones de los demás jugadores). El equilibrio de Nash tiene lugar cuando todos los jugadores aplican estrategias de mejor respuesta (ver [10]). Al igual que en el caso del óptimo social, la técnica usada es inducción hacia atrás, calculando el equilibrio de Nash en cada etapa del juego. Es importante notar que en este caso el *supuesto de sobreconsumo* se aplica en el último período del juego (una vez que tal período finaliza, no hay juego que jugar, por tanto los jugadores tratarán de consumir tanto como sea posible). Procediendo como en el caso del óptimo social, se encuentra la siguiente expresión para las fracciones de consumos:

$$f^*(T, \alpha, \delta, n) = \frac{1}{(n-1) + \sum_{i=0}^{T-1} (\alpha\delta)^i}$$

Solución de Nash

### 2.4 Comparación del óptimo social y el equilibrio de Nash

Para mostrar que ocurre una tragedia se comparan la solución de Nash y el óptimo social. Es necesario tener presente que cuanto mayor sea el consumo presente, menor será el recurso disponible en períodos futuros.

En este sentido, la solución de óptimo social implica un balance entre los consumos inmediatos y los de largo plazo, al contrario de la solución de Nash.

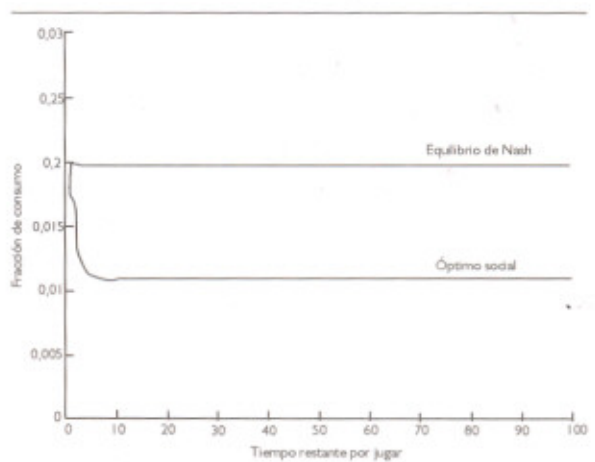


Figura 1. Comparación de óptimo social y equilibrio de Nash<sup>2</sup>.

La figura 1 muestra que la solución de Nash es casi un valor constante en términos de  $T$ . En contraste, el óptimo social es restringido si se compara con el equilibrio de Nash, sólo cuando se acerca el final del juego empieza a aumentar la cantidad de consumo. Esto tiene sentido, dado que cuando el juego está a punto de terminar, no hay mucha razón para restringirse para "ahorrar" para el futuro, pues ya pronto el juego se acabará.

El comportamiento restringido de la estrategia de juego de óptimo social contribuye a alcanzar la sostenibilidad del recurso, lo que puede ser demostrado formalmente al reemplazar las expresiones de consumo generadas por el óptimo social y el equilibrio de Nash en la ecuación que regula la dinámica del recurso. Esto da lugar a un sistema dinámico discreto que representa el comportamiento del recurso consumido. Para el óptimo social, el recurso alcanza un estado sostenible reflejado en un punto fijo estable diferente de cero. Por otro lado, para la solución de Nash el sistema tiene un punto fijo estable mucho más cercano a cero (ver [9]).

2. En adelante, cuando se comparen reglas y se hagan cálculos sólo en términos de  $T$ , los valores fijos considerados de  $\alpha$  y  $\delta$  serán 0,5 y 0,9 respectivamente.

### 3. MODELO DE SIMULACIÓN

Esta sección presenta el modelo del juego de simulación. La técnica usada es Programación Genética (PG), un tipo de algoritmo evolutivo (AE) en el cual cada individuo es un programa de computador, en particular una función matemática [11]. Cada función es construida como la composición de elementos de un conjunto bien definido de terminales (argumentos para la función) y primitivas (otras funciones fundamentales que constituyen cada individuo). Los individuos son evaluados de acuerdo con una función de *fitness*; el proceso evolutivo trata de favorecer a aquellos individuos cuyo valor de *fitness* es alto.

La dinámica generada por la PG es una secuencia de conjuntos de programas que puede interpretarse como la evolución de una sociedad artificial donde cada agente es representado como un programa [12]. Las expresiones matemáticas generadas con PG representan estrategias de juego similares a las descritas en la Sección 2, donde cada regla determina las acciones de un jugador durante el juego.

Empezando con una población de estrategias de juego creadas aleatoriamente, las nuevas estrategias se construyen a partir de las anteriores de manera evolutiva. Las reglas de juego pertenecen al conjunto unión de las primitivas y terminales, denominado el espacio de búsqueda. Por otro lado, la función de *fitness* conduce el proceso de búsqueda. Estos elementos se describen a continuación.

#### 3.1 Representación genotípica

El espacio de búsqueda debe corresponder a un conjunto de estrategias de juego similares a las predichas por la teoría de juegos clásica. Por tanto, es necesario diseñar las terminales y las primitivas de forma que permitan al proceso evolutivo alcanzar reglas como el equilibrio de Nash y el óptimo social.

Así, el conjunto de terminales  $\tau$  es igual a  $\{\alpha, \delta, T, n, l\}$ , donde  $\alpha$  es el valor constante asociado a la tasa de renovación del recurso explotado;  $\delta$  es el factor de descuento;  $T$  es el número de interacciones restantes para concluir un juego determinado;  $n$  es el número de jugadores en el juego, y  $l$  es el valor constante.

Por otro lado, el conjunto de primitivas  $\mathcal{F}$  es igual a  $\{/, *, +, \Sigma^{\wedge}\}$  donde  $+, /, *$  son las operaciones aritméticas, suma, división, y producto, respectivamente, y  $\Sigma^{\wedge}$  es una función llamada suma geométrica, definida como

$$\Sigma : (\mathbb{Z}^+ \times \mathbb{R}) \rightarrow \mathbb{R}$$

$$\hat{\Sigma}(n, x) \mapsto \sum_{i=0}^{n-1} x^i$$

Como el primer parámetro de la primitiva suma geométrica no es un valor real, no hay clausura entre operadores (ver [11]), de forma que la implementación de PG está diseñada para mantener reglas sintácticamente válidas evitando aquellas donde el primer parámetro de la suma geométrica no está en  $\mathbb{Z}^+$ .

Las funciones que corresponden a las estrategias de juego se representan genéticamente como cadenas de símbolos en  $\tau \cup \mathcal{F}$ . Para que la regla de consumo se asocie a una cadena (cromosoma), se usa una pila de valores numéricos, como se muestra en la figura 2. Esto se conoce como PG basada en pilas (ver [13]).

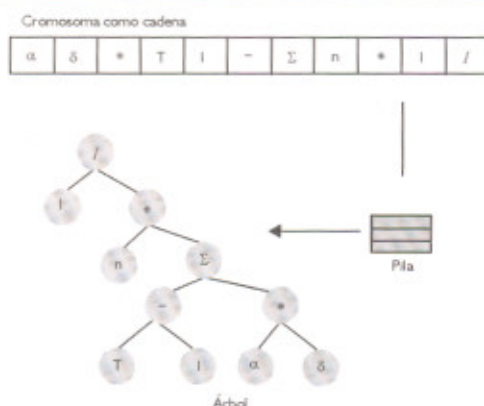


Figura 2. Regla de óptimo social representada como cadena y luego decodificada como árbol usando una pila.

#### 3.2 Representación fenotípica

En esta sección la pregunta a abordar es: Dada una función que representa la estrategia de un jugador, ¿cómo se calcula el consumo a lo largo del juego? Una estrategia de juego (regla) construida a partir de las primitivas y terminales ya discutidas, determina qué fracción del recurso disponible debe consumir un jugador, dados los valores de  $T, n, \alpha$  y  $\delta$ .

Sea  $g$  la función usada por un agente  $i$  para jugar un juego particular. Claramente, debido a la definición de primitivas y terminales, esta función tiene como rango la recta real. Sin embargo, se supone que las funciones otorgan fracciones para cualquier conjunto de valores de los parámetros; por tanto, siempre que una función no esté en  $[0, 1]$  en una etapa específica del juego, la regla se considera semánticamente inválida, y el consumo asignado es cero.

En adelante, sea  $G$  la suma de las funciones  $g$  de todos los jugadores en un período particular

$$G = \sum_{i=1}^n g^i(\cdot)$$

Considerando la población como un todo, aun asumiendo que todas las estrategias de juego son semánticamente válidas, la posibilidad de sobreconsumo sigue existiendo. Esto sucederá siempre que, en un estadio determinado del juego,  $G$  sea mayor que 1. En este caso se aplica el supuesto de sobreconsumo. Por tanto, el consumo asignado (para aquellos jugadores que estén usando reglas semánticamente válidas) es el recurso disponible en el momento dividido igualmente entre los jugadores.

En resumen, si  $c_{it}$  es la cantidad de recurso consumida por un agente  $i$  en un período  $t$  de un juego, entonces  $c_{it}$  se calcula como

$$c_{it} = \begin{cases} y_t g^i(T, \alpha, \delta, n) & \text{if } G \leq 1 \text{ y } g^i(\cdot) \in [0, 1] \\ \frac{y_t}{n} & \text{if } G > 1 \text{ y } g^i(\cdot) \in [0, 1] \\ 0 & \text{if } g^i(\cdot) \notin [0, 1] \end{cases}$$

donde  $y_t$  es la cantidad de recurso disponible para tal período. En el primer caso, el resultado dado por la regla es una fracción, y la población no sobreconsume; este se puede considerar el caso ideal, cuando todos los jugadores son semánticamente válidos. El segundo caso tiene en cuenta el supuesto de sobreconsumo. El último caso es consecuencia de la posibilidad de invalidez semántica, jugadores cuya regla no tiene sentido y no se les permite ningún consumo.

De acuerdo con esto, el valor de juego (*payoff*) de un jugador en un estadio determinado de un juego se computa como el logaritmo de su consumo. Adicio-

nalmente, la dinámica del recurso para un juego se computa de acuerdo con la ecuación especificada en el marco matemático.

### 3.3 Evaluación de la función de *fitness*

El objetivo del proceso de adaptación es favorecer a los jugadores más aptos. El principal criterio para dicha aptitud es el *valor de juego* obtenido por los jugadores del juego, así es necesario tener en cuenta el valor de juego obtenido por los jugadores a lo largo de todo el juego en la función de *fitness*.

La medida de *fitness* considerará dos cantidades adicionales. Como se mostró anteriormente, existen dos estados no deseables para una regla: la invalidez semántica y el comportamiento de sobreconsumo. La primera, como se explica en [12], es un problema común cuando se aplican técnicas de PG a este tipo de situaciones. La segunda está también presente en el modelo clásico (es doblegada al introducir una restricción de optimización - ver ecuación 1). Así, el mecanismo usado para tratar dichos estados indeseables es apartarlos del proceso evolutivo por medio de la función de *fitness*, que considera incluye dos valores adicionales, uno que se ocupa de la invalidez semántica y otro que considera el comportamiento de sobreconsumo.

En resumen, la medida de *fitness* tiene tres metas:

1. Conservar a los individuos cuyos valores de juego son altos.
2. Remover a los individuos que no son semánticamente válidos.
3. Descartar a los individuos que tienden a sobreconsumir.

La manera de alcanzar tales metas se discute a continuación.

#### 3.3.1 Contribución de valor de juego (*payoff*)

Suponga que un jugador  $i$  completa un juego que consta de  $L$  períodos. El valor de juego recibido por tal jugador en una etapa  $t$  está determinado por

$$u_t^i = \log c_t^i$$

donde  $c_{it}$  es el recurso consumido por el jugador en tal período. Al final del juego, el valor de juego recibido por tal agente es

$$U^i = \sum_{t=1}^L u_t^i$$

La cantidad  $U^i$  proporciona una medida de la contribución del valor de juego al fitness. En aras del tratamiento computacional, esta cantidad es llevada a un número  $p^i$  en  $[0, 1]$ . Este hecho es más un asunto de implementación (para este propósito se usa una transformación lineal debidamente parametrizada).

### 3.3.2 Penalización por invalidez semántica

Considérese una función indicadora de la forma.

$$I(x) = \begin{cases} 1 & \text{si } x \in [0, 1] \\ 0 & \text{en los demás casos} \end{cases}$$

Sea  $f_t^i$  la estrategia usada por un jugador en un determinado juego compuesto de  $t = 1, \dots, L$  períodos. La invalidez semántica de un período se calcula entonces como

$$d^t(f^i(\cdot)) = I(x) \min(|f^i - 1|, |f^i|)$$

Claramente,  $d^t(f^i(\cdot))$  es alto cuando el resultado arrojado por la regla se aleja de  $[0, 1]$ , y cero cuando el resultado está en  $[0, 1]$ . En consecuencia, para un juego completo, la medida de invalidez semántica se resume en un número  $D^i$ , definido como

$$D^i = \sum_{t=1}^L d^t(f^i(\cdot))$$

En aras de la tratabilidad computacional,  $D^i$  es llevado a un número  $s_i$  en  $[0, 1]$ , usando una transformación lineal tal que (al final del juego) el agente cuyo  $D^i$  es mayor recibe 1, y aquel cuyo  $D^i$  es menor recibe 0. Así, al final del juego, un número  $s_i$  proporciona una medida de la invalidez semántica de cada uno de los agentes durante el juego completo.

### 3.3.3 Penalización del comportamiento de sobreconsumo

Si  $g_t^i$  es la estrategia usada por un agente  $i$  en un juego determinado de  $L$  períodos, una medida de la tendencia al sobreconsumo de tal agente para un período  $t$  puede calcularse como:

$$q^t(g^i(\cdot)) = \begin{cases} 0 & \text{si } G \leq 1 \\ (1-G) \frac{g^t(\cdot)}{G} & \text{en los demás casos} \end{cases}$$

Así, para un juego completo la cantidad que mide la tendencia al sobreconsumo del jugador  $i$  es

$$o^i = \sum_{t=1}^L q^t(g^i(\cdot))$$

Esta cantidad es transformada linealmente en  $[0, 1]$ , de nuevo asignando 1 al mayor valor y 0 al menor valor.

### 3.3.4 Función de fitness

Después de terminado el juego, la adaptabilidad de un jugador  $i$  se representa como un vector

$$v^i = (p^i, s^i, o^i)$$

y su medida de fitness será una «escalarización» del mismo. Para los propósitos de este trabajo, los valores de fitness de los agentes se definen entonces como

$$F^i(v^i) = p^i (1 - \max(s^i, o^i))$$

La idea de esta función es evitar estrategias semánticamente inválidas y mantener la esencia de la asignación de valor de juego en el modelo analítico. La función  $\max$  (máximo) se usa para aplicar un castigo fuerte sobre la invalidez semántica y la violación de restricciones. Entonces, se espera tener individuos cuyo fitness es sólo proporcional al valor de juego recibido en el juego a lo largo del proceso evolutivo. Esta es una función de fitness multiobjetivo (ver [14] para más detalles), dado que puede considerarse que el proceso tiene tres metas: maximizar el valor de juego mientras se minimiza la invalidez semántica y la tendencia al sobreconsumo.

### 3.4 El algoritmo evolutivo

Considérese un procedimiento  $\mathcal{D}(P, T_{M\acute{A}X}, Y_0, \alpha, \delta)$  que simula un juego y computa los valores de *fitness*, de acuerdo con el desempeño obtenido por la población en el juego, como se explicó antes. Los parámetros de este procedimiento son:

- Una población  $P$  de estrategias de juego ( $|P|$  denota el número de jugadores)
- Un valor real  $Y_0$ , que representa la cantidad de recurso disponible al comienzo del juego
- Un valor entero  $T_{M\acute{A}X}$ , el número de etapas del juego
- La tasa de renovación del recurso  $\alpha$  y el factor de descuento  $\delta$ .

Así, el algoritmo opera de la siguiente manera

```

1  i ← 0
2  Pi ← RANDOM INITIAL POPULATION (·)
3  while stopping criterion not met
4    do assign TM\acute{A}X, Y0, α, δ al random
5      D(Pi, TM\acute{A}X, Y0, α, δ)
6      Ps ← RANKING-SELECTION (Pi)
7      Pc ← Crossover (Ps)
8      Pm ← MUTATION (Pc)
9      i ← i + 1
10     Pi ← Pm
11  return Pi

```

Algoritmo evolutivo

## 4. RESULTADOS EXPERIMENTALES

Se desarrolló un conjunto de experimentos, donde un experimento se consideraba como una corrida del algoritmo evolutivo descrito arriba. El criterio de terminación del algoritmo es el número deseado de generaciones. La salida de un experimento es el conjunto de estrategias de juegos obtenidas por el algoritmo evolutivo en la última generación. Intuitivamente, un experimento puede considerarse como el proceso por el cual una sociedad se adapta a través de las generaciones para alcanzar una política de explotación del recurso apropiada.

Como el óptimo social y la regla de Nash tienen respectivamente una representación en árbol de 11 y 13

nodos, el tamaño máximo de un árbol se limitó a 14 nodos. Las variables ambientales, es decir  $T_{M\acute{A}X}$ ,  $Y_0$ ,  $\alpha$  y  $\delta$ , fueron generadas aleatoriamente para cada juego simulado, con el propósito de confrontar las estrategias de juego en adaptación con un número considerable de condiciones ambientales para alcanzar reglas generales.

La selección se llevó a cabo mediante el método de *selección por torneo elitista* de tamaño cuatro (ver [14] para más detalles). Además se usó *cruce simple* (*crossover*) de un solo punto. Adicionalmente, un índice aleatorio de mutación (distribuido de manera uniforme en  $[0, 1]$ ) emuló el comportamiento innovador de los agentes, asumiendo que la propensión de los agentes a "experimentar" con nuevas reglas es una variable aleatoria que cambia con las generaciones.

El algoritmo evolutivo de programación genética basada en pila fue implementado en lenguaje de programación Java, los resultados experimentales fueron almacenados en archivos de texto, y posteriormente analizados estadísticamente usando Matlab. Dado el alto costo computacional del proceso de experimentación, se utilizó un esquema de computación distribuida basado en JavaSpaces.

En total se llevaron a cabo 225 experimentos. El tamaño de la población considerada fue 50 individuos. En cada experimento, el algoritmo se corrió por 1500 generaciones, lo que, empíricamente y bajo las condiciones estudiadas, resultó ser un número apropiado de pasos para alcanzar la estabilización completa dentro de la población, es decir, se obtuvieron estrategias estables simétricas. Este estado de variación no fenotípica puede considerarse un estado de estabilidad evolutiva en el sentido de Riechmann [15].

En los experimentos, el 91 por ciento de las reglas obtenidas fueron semánticamente válidas. Por lo tanto, la tarea de evitar las reglas semánticas inválidas se alcanzó exitosamente. Al conservar sólo las reglas semánticamente válidas, la muestra experimental consistió de 205 reglas finales.

En términos del valor de *fitness*, la mejor regla hallada fue  $g(\cdot) = \frac{\alpha}{n}$  que fue también la moda dentro de la muestra, siendo la regla usada por todos los jugadores en el 5 por ciento de los experimentos. La figura 3 muestra una comparación gráfica de esta regla contra las de Nash y del óptimo social, en términos de  $T$ .



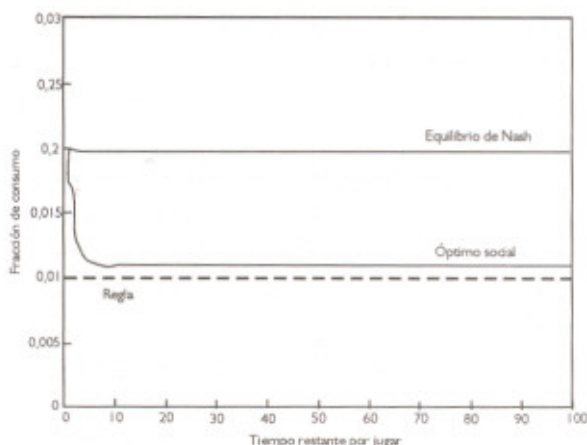


Figura 3. comparación gráfica de la regla más apta contra el equilibrio de Nash y el óptimo social.

Salta a la vista que esta regla se encuentra por debajo del equilibrio de Nash, y considerablemente más cerca del óptimo social. Además, dicha regla es aún más restringida que el óptimo social; por tanto, implica un mejor nivel de sostenibilidad para la dinámica del recurso.

La regla correspondiente a la mediana del valor de fitness fue  $g(\cdot) = \frac{\alpha^3}{n}$ . La figura 4 muestra la comparación gráfica de esta regla contra el equilibrio de Nash y el óptimo social. Nótese que esta regla se encuentra casi en el medio, entre la regla de Nash y el óptimo social.

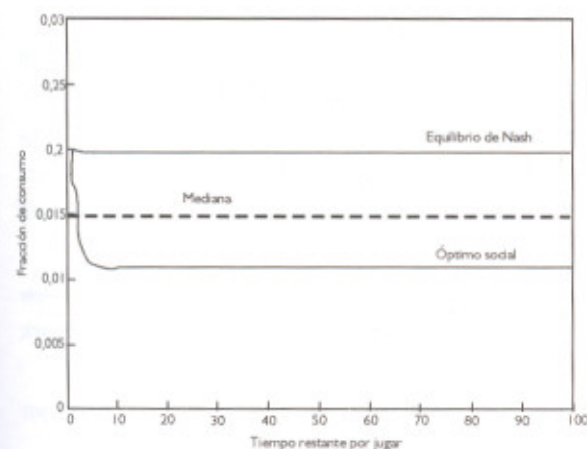


Figura 4. Regla mediana en comparación con las reglas óptimo social y de Nash.

Como las soluciones teóricas de Nash y el óptimo social son las reglas de referencia, es necesario definir un criterio numérico para arrojar luz sobre la proximidad de las soluciones obtenidas y las predicciones de la teoría de juegos clásica. Así, se calculará la distancia entre las estrategias evolucionadas y dichas funciones (i.e., Nash y la solución de óptimo social). Las reglas cercanas a la función de la solución de óptimo social pueden considerarse estrategias de juego cooperativas, al contrario de aquellas próximas al equilibrio de Nash.

Como todas las reglas representan fracciones, es posible usar una métrica en el espacio de las funciones en  $[0, 1]$  (ver [16] para más detalles), como sigue. Sea  $A$  el dominio de dos funciones  $x$  y  $y$ , entonces, la distancia entre ellas se calcula como

$$d(x, y) = \sup_{t \in A} |x(t) - y(t)|$$

Específicamente, estas distancias entre las reglas obtenidas y las de referencia (el equilibrio de Nash y la solución de óptimo social) se calcularon usando un algoritmo de aproximación numérica. En la figura 5 se muestra un *boxplot* de dichas distancias.

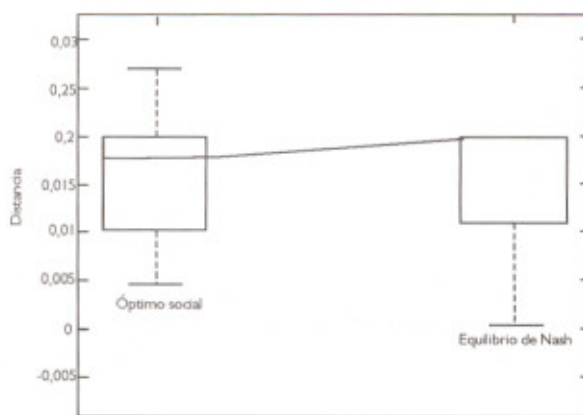


Figura 5. *Boxplot* de la distancia de las reglas coevolucionadas respecto a las soluciones social y de Nash.

Nótese que las dos «cajas» son prácticamente iguales. En particular, la distancia al rango del cuartil interior del óptimo social es 0.0098; entre tanto, la distancia al

rango intercuartílico de Nash es 0.0090. En otras palabras, el cincuenta por ciento de las distancias (desde las estrategias de juego evolucionadas hasta el equilibrio de Nash y el óptimo social) se encuentran en el mismo rango. Esto significa que la cooperación (representada por las reglas cercanas al óptimo social) y la no cooperación (representada por las reglas cercanas a Nash) pueden emerger con aproximadamente la misma probabilidad.

Asimismo, la distancia de la mayoría de las reglas evolucionadas al equilibrio de Nash se encuentra por debajo de la regla correspondiente a la mediana (note que la mediana de las distancias a la solución de Nash es básicamente igual a la distancia máxima obtenida para todas las reglas). Además, como los valores de las dos medianas son muy cercanos (la mediana de la distancia al óptimo social es 0.01775 y la mediana de la distancia al equilibrio de Nash es 0.01967), la mayoría de las reglas evolucionadas están a la misma distancia tanto del equilibrio de Nash como del óptimo social.

## 5. CONCLUSIONES

En este trabajo, se usó un juego dinámico como marco para estudiar las relaciones entre la adaptación y la cooperación. Específicamente, se usó PG para evolucionar estrategias de juego en un escenario de tragedia de los comunes. Aunque el ambiente dinámico parece evitar la conquista de reglas completamente cooperativas en todos los casos, se mostró que la cooperación —reflejada en la regla de óptimo social— puede emerger, evitando la tragedia. Particularmente, en los experimentos realizados se observó que la cooperación (representada por las reglas cercanas al óptimo social) y la no cooperación (representada por las reglas cercanas a Nash) emergieron con aproximadamente la misma probabilidad.

Por otro lado, en los experimentos se obtuvieron algunas estrategias de juego que estaban por debajo del equilibrio de Nash, lo que muestra que la tragedia de los comunes puede resolverse por medio de la adaptación. Específicamente, para evitar la tragedia, los jugadores cooperan al restringir sus propios consumos (ahorrando para el futuro), y al jugar estrategias que están por debajo del equilibrio de Nash, lo que implica un nivel mayor de sostenibilidad del recurso. Adicio-

nalmente, la mayoría de las reglas evolucionadas se encontraron (estadísticamente) aproximadamente a la misma distancia tanto del equilibrio de Nash como del óptimo social.

Sin embargo, quedan muchas preguntas por responder; en particular, sería valioso considerar otras representaciones de las estrategias de juego dentro de este marco, y analizar no sólo las poblaciones estables finales, sino también el proceso por el cual se alcanzan estos estados. Se requiere mucho trabajo para evaluar la sensibilidad de la cooperación con respecto a la variación de los parámetros de los algoritmos evolutivos.

## REFERENCIAS

- [1] Hardin, G. (1968). The tragedy of the commons, *Science*, vol. 162, pp. 1243-48.
- [2] Schelling, T. C. (1978). *Micromotives and Macrobehavior*. W.W Norton and Co.
- [3] Aliprantis, C.D. (2000). *Games and Decision Making*. Oxford University Press.
- [4] Yao, X. (1997). Automatic acquisition of strategies by co-evolutionary learning. Blumenstein [153], pages 23-29.
- [5] Fogel, D.B. (1995). *Evolutionary Computation: toward a new philosophy of machine learning*. New York, NY: IEEE Press.
- [6] Darwen, P.J., and Yao, X. (2001). Why more choices cause less cooperation in iterated prisoner's dilemma. In *Proceedings of the 2001 Congress on Evolutionary Computation CEC2001* (COEX, World Trade Center, 159 Samseong-dong, Gangnam-gu, Seoul, Korea), pp. 987-994, IEEE Press, 27-30.
- [7] Yao, X., and Darwen, P.J. (1994). An experimental study of  $n$ -person iterated prisoner's dilemma games. In *Evo Workshops*, pp. 90-108.
- [8] Deadman, S.E.; Peter, J., and Randy, G. (2000). Simulating common pool resource management experiments with adaptive agents employing alternate communication routines. *Journal of Artificial Societies and Social Simulation*, vol. 3, no. 2.
- [9] Levhari, D., and Mirman, L. (1980). The great fish war: An example using a dynamic Cournot-Nash solution, *The Bell Journal of Economics*, vol. 11, no. 1, pp. 322-34.
- [10] Dutta, P.K. (1999). *Strategies and Games: Theory and Practice*. MIT Press.
- [11] Koza, J.R. (1992). *Genetic programming: On the programming of computers by natural selection*. Cambridge, Mass.: MIT Press.

- [12] Chen, S.H. (2001). Fundamental issues in the use of genetic programming in agent-based computational economics. *Working Paper, AI-Econ Research Center National Chengchi University Taipei, Taiwan.*
- [13] Perkis, T. (1994). Stack-based genetic programming. In *Proceedings of the 1994 IEEE World Congress on Computational Intelligence*, vol. 1, (Orlando, Florida, USA), pp. 148-153, IEEE Press, 27-29.
- [14] Fogel, D.; Back, T.; Michalewicz, Z. (Eds.) (1997). *Handbook of Evolutionary Computation*. IOP Publishing Ltd and Oxford University Press.
- [15] Riechmann, T. (2001). Genetic algorithm learning and evolutionary games. *Journal of Economic Dynamics and Control*, no. 25, pp. 1019-1037.
- [16] Kreyszig, E. (1978). *Introductory Functional Analysis with Applications*. New York, NY, USA: John Wiley & Sons.