Un Acercamiento con Aprendizaje Reforzado a Monopolio

Carlos Coronado[1] Mathias Nieva[2] Pablo Rivas[3] & Judá Villalta[4]

Universidad Privada Boliviana
Facultad de Ingenierias y Arquitectura

Introdución

En 1796 Wolfgang von Kempelen creó un autómata llamado **El Turco**, el cual podía jugar ajedrez a un nivel maestro, aunque en realidad este era controlado por una persona en su interior. Sin embargo, quedó la duda de si una máquina podría superar a un humano en dicho juego. Esta duda sería respondida en 1996 con la llegada de Deep Blue.
En la actualidad, con la inteligencia artificial se ha logrado extrapolar este éxito a otros juegos de mayor complejidad, los cuales, sin la IA serían imposibles para una máquina, siendo Alpha Go el mejor ejemplo. El aplicar el aprendizaje reforzado (RL) en juegos nos permite recrear un entorno en el cual el agente inteligente es capaz de aprender con prueba y error para mejorar su comportamiento y alcanzar un resultado óptimo.
En la presente se expone una aplicación de RL para Monopolio, juego cuyo desarrollo suele ser impredecible por momentos y que requiere de una correcta administración de los recursos disponibles por cada jugador.

Metodología

Este trabajo utiliza el enfoque de RL para modelar el juego Monopolio como un proceso de decisión de Markov (MDP), dado a que el número de posibles estados en los que puede estar el juego es muy numeroso se determina usar representaciones de aspectos importantes del juego junto con redes neuronales, esto permite a los agentes RL aprender estrategias para ganar. Se implementó dos agentes básicos de Q-Learning, junto con el entorno, cada uno con un sistema de recompensas distinto. Se comparó su desempeño junto con dos agentes de referencia, uno que utiliza una política aleatoria y otro que utiliza una política fija basada en el dinero que posee.
Nuestra propuesta para el modelo de recompensas es la siguiente:

$$r(x) = \frac{x}{1 + |x|}$$

$$x = asset_{factor} + finance_{factor}$$

$$asset_{factor} = \begin{Bmatrix} j - \mu_p \ ; \ si \ c < 1 \\ (j - \mu_p) + (\frac{h}{5*c}) \ ; \ si \ c \geq 1 \end{Bmatrix}$$

$$finance_{factor} = d_j - \mu_{op}$$

$j \rightarrow \#\ de\ propiedades\ del\ jugador$
$\mu_p \rightarrow media\ del\ \#\ de\ prop\ de\ los\ oponentes$
$c \rightarrow \#de\ colores\ completados$
$h \rightarrow \#\ de\ casas\ construidas$
$d_j \rightarrow dinero\ del\ jugador$
$\mu_{op} \rightarrow media\ del\ dinero\ de\ los\ oponentes$

El otro modelo es el que se encuentra descrito en [1], paper sobre el que se basa este trabajo.
La implementación se realizó en el lenguaje C#, se trabajó entrenando por 400 juegos, 600 menos que en el trabajo original [1], a los cuatro agentes propuestos en conjunto para posteriormente realizar diferentes experimentos y así evaluar el rendimiento del agente que implementa nuestra propuesta de modelo de recompensas.

Resultados

Los experimentos fueron realizados efectuando 100 partidas entre los agentes a evaluar.
Para el primer experimento se enfrentaron los dos agentes basados en Q-Learning, el agente 2 corresponde a la implementación original [1], mientras el agente 1 utiliza nuestra propuesta de modelo de recompensa. En la figura 1 puede verse el desarrollo de victorias de cada agente y, como se puede apreciar, nuestra propuesta supera ampliamente al modelo original de rewarding.

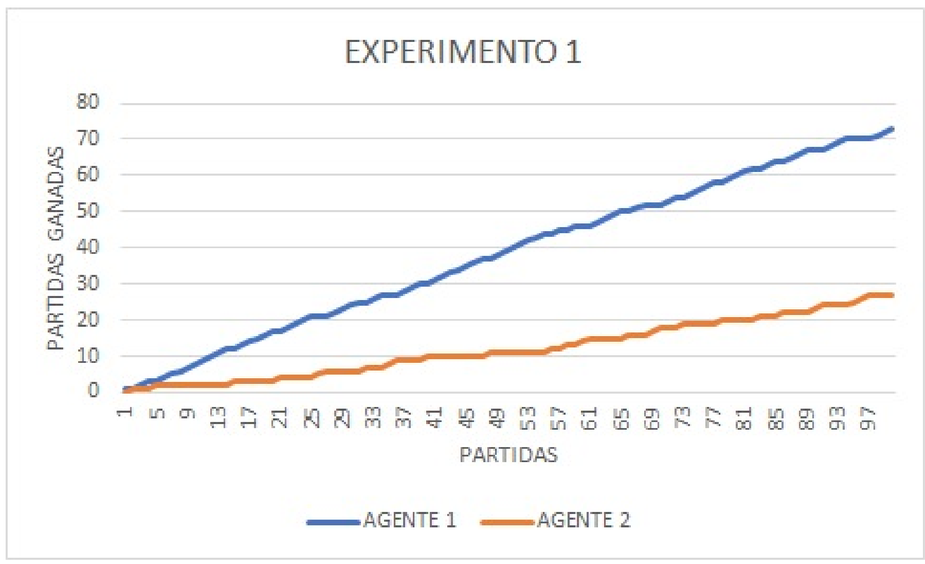

Figura: Número de victorias de cada agente inteligente

Para el segundo experimento se enfrento el agente inteligente, que utiliza nuestra propuesta de modelo de recompensa, contra los restantes de referencia igualmente utilizados en el trabajo original [1]. Para los agentes de referencia antes mencionados se consideran los siguientes puntos:

- El agente de política aleatoria selecciona sus acciones aleatoriamente e ignorando la señal de estado.
- El agente de política fija, toma sus decisiones en función al dinero que posee. Si tiene menos de 150 vende, si tiene más de 350 compra y si está en el rango de 150 a 350 no hace nada.

En la figura 2 se puede ver el desarrollo de victorias de cada agente durante el experimento, mostrando al agente inteligente como un amplio ganador.

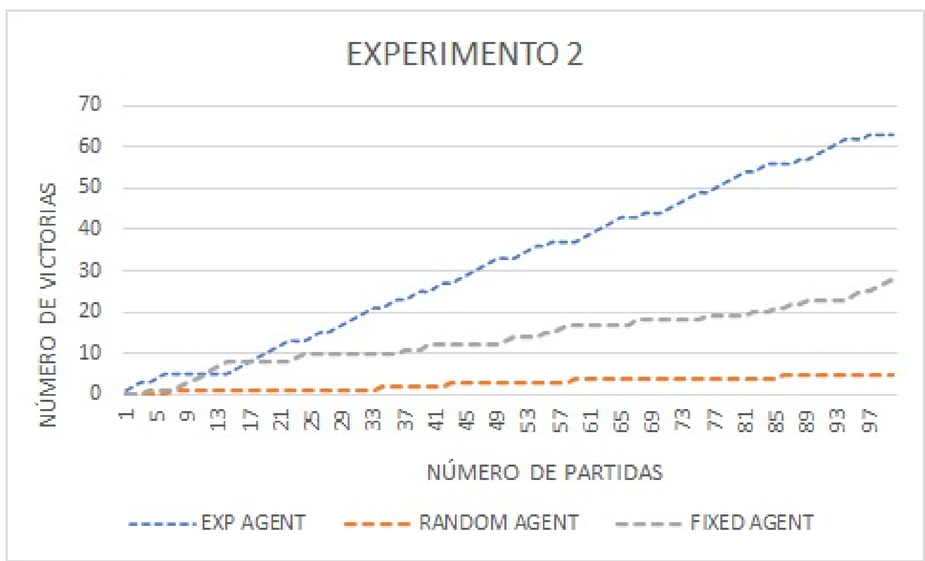

Figura: Número de victorias de cada agente en 100 partidas

Conclusión

Los resultados obtenidos muestran que el nuevo modelo de recompensa mejora el accionar del agente inteligente. Esto no necesariamente significa que sea un modelo que provea de una política superior a la original, hay la posibilidad de que lo que estén mostrando los resultados es que el nuevo modelo planteado converja más rápidamente a una política óptima, puesto a que no se vió un estancamiento en el aprendizaje en alguno de los dos modelos. Aún así, el que converja más rápidamente es un punto a favor muy importante.

References

Panagiotis Bailis, Anestis Fachantidis y Ioannis Vlahavas. "Learning to play Monopoly: A Reinforcement Learning approach". en. En: (), pág. 3.