



Trabajo Fin de Grado

Título: Diseño de Estrategias Óptimas en el Póker mediante Algoritmos Genéticos.

Autor: Gabriel Marco Ángeles.

Titulación: Grado de Ingeniería de Sistemas Audiovisuales.

Tutor: José Miguel Leiva Murillo.

Fecha: 03/09/2012



Índice

1. Introducción	3
1.1 Póker Texas Hold´Em	4
1.2 Marco Regulador	7
1.3 Marco socioeconómico	8
2. Algoritmos Genéticos	9
2.1 Descripción de los Algoritmos Genéticos	9
2.2 Teoría de los Juegos	18
2.3 El criterio de <i>Kelly</i>	19
2.4 Probabilidades en el Póker, aplicación del criterio de <i>Kelly</i>	20
3. Diseño de la solución técnica	23
3.1 Diseño de la estrategia	23
3.2 Codificación de la baraja	31
3.3 Identificación de la jugada	31
3.4 Diseño de la función <i>fitness</i>	35
3.5 Diseño del Algoritmo genético	40
4. Resultados y conclusiones	45
4.1 Convergencia	45
4.2 Resultados	51
4.3 Conclusiones	60



1. INTRODUCCIÓN

Este trabajo tiene como objetivo principal diseñar estrategias óptimas en el Póker (modalidad Texas Hold´Em) mediante Algoritmos Genéticos.

El Póker Texas Hold´Em es un juego de cartas cuyo objetivo principal es, para todos los jugadores, formar la mejor jugada o combinación de cinco cartas posible con dos cartas individuales y cinco que son comunes. Los jugadores apostarán en función de sus naipes cuando lo consideren oportuno.

Se define una estrategia como un conjunto de probabilidades de llevar a cabo acciones ante situaciones determinadas, en este caso, relacionadas con el Póker. Encontrar una estrategia óptima se considera un problema de optimización numérica muy complejo, para ello se utilizan los Algoritmos Genéticos.

Los Algoritmos Genéticos son métodos adaptativos que se emplean para la resolución de problemas de búsqueda y optimización. Permiten optimizar funciones numéricas (llamadas *fitness*), generalmente aplicadas sobre datos binarios cuya naturaleza analítica es muy compleja o se desconoce por completo y su búsqueda exhaustiva es imposible de realizar.

Una vez caracterizada la estrategia de un jugador se obtiene su valor de *fitness*, correspondiente al resultado de una partida en la que compiten varios individuos, cuyo valor determinará la generación de nuevos individuos en el Algoritmo Genético.

El proceso de optimización imita al que tiene lugar en los procedimientos biológicos evolutivos, en el cual una población inicial de individuos sufre procesos como cruce (reproducción) y mutación originando una nueva población de individuos mejores (en este caso, soluciones a la función *fitness* a optimizar) que sustituirán a sus predecesores, continuando así el Algoritmo Genético de forma iterativa hasta lograr su convergencia, lo que permite hallar una solución final.

Los Algoritmos Genéticos son muy potentes y efectivos, sobre todo para solucionar problemas con un espacio de búsqueda para los que no existe una técnica específica para resolverlos o es muy difícil de implementar.

Para la resolución del problema planteado se utiliza el software matemático MATLAB ya que brinda grandes prestaciones para el cálculo numérico de una manera eficiente y precisa.



1.1 EL PÓKER TEXAS HOLD'EM





Refs: [1],[2] y [12]

El Texas Hold'Em es la modalidad del póker más extendida y jugada en los casinos de todo el mundo. En sus torneos pueden llegar a participar miles de jugadores en mesas de 8 a 10 jugadores normalmente. Aunque tiene reglas sencillas, su estrategia es muy compleja.

El objetivo de este juego consta en conseguir la mejor combinación de cinco cartas posible con dos cartas que cada jugador recibe individualmente, denominadas *pocket cards* y cinco cartas expuestas en la mesa que son comunes para todos los jugadores.

La baraja empleada es de tipo inglés de 52 cartas, cuyo valor de mayor a menor es el siguiente: As, rey (K), dama (Q), jota (J), diez, nueve, ocho, siete, seis, cinco, cuatro, tres, dos.

Sus cuatro palos (corazones, picas, diamantes y tréboles) tienen igual preferencia para cualquiera de las combinaciones posibles que se pueden conseguir. Las combinaciones son las siguientes, ordenadas de mayor a menor:

	Jugada	Descripción	Ejemplo	Combinaciones posibles
1	Escalera real o flor imperial	Cinco consecutivas seguidas del mismo palo del As al 10.		4
2	Escalera de color	Cinco cartas consecutivas del mismo palo, en caso de empate decide la carta más alta.		36
3	Póker	Cuatro cartas de igual valor. En caso de empate, gana el Póker con cartas más altas.		624
4	Full	Combinación de tres cartas iguales (trío) y dos cartas iguales (par). En caso de empate gana el trío más alto		3.774



		y luego el par más alto.		
5	Color	Cinco cartas del mismo palo. En caso de empate, gana la carta más alta.		5.108
6	Escalera	Cinco cartas consecutivas de palos diferentes. En caso de empate, gana la escalera que tenga la carta de valor más alto.		10.200
7	Trío	Tres cartas de igual valor. En caso de empate gana el trío más alto.		54.912
8	Dobles Parejas	Combinación de dos pares de cartas. En caso de empate decide la pareja más alta. Si continúa el empate lo hará la pareja más baja.		123.552
9	Pareja	Dos cartas de igual valor. En caso de empate, decide la pareja más alta.		1.098.240
10	Carta Alta	Carta de mayor valor. En caso de empate decide la siguiente carta más alta y así sucesivamente.		1.303.596

Figuras: Póker Texas Hold´Em www.wikipedia.es



Texas Hold´Em sin Límite

Esta modalidad de juego es la empleada en el proyecto y la más común entre todas las modalidades. Recibe este nombre debido a que no hay una apuesta máxima. Cada jugador puede arriesgar todo el dinero que posea si así lo desea. El resto de normas, que se explican a continuación, son iguales a las de las modalidades Texas Hold´Em con Límite y Texas Hold´Em Pot-Limit.

Se comienza con las *ciegas*, que son dos apuestas obligatorias (*ciega pequeña* y *ciega grande*) para los jugadores que se encuentran a izquierda del repartidor. El primer jugador a su izquierda apostará la *ciega pequeña*, y el siguiente la *ciega grande*. La *ciega pequeña* es equivalente a la mitad de la apuesta mínima y la *ciega grande* a la apuesta mínima.

Primera Fase. Preflop:

Una vez realizadas estas dos primeras apuestas obligatorias, el repartidor da dos cartas boca abajo a cada jugador (a lo largo de la mano se repartirán cinco cartas boca arriba, llamadas cartas comunitarias, que son comunes para todos los jugadores).

El objetivo es formar la mejor combinación de cinco cartas entre dos de sus cartas y las cinco comunitarias, pudiendo usar dos cartas de su mano, una o ninguna.

Comienza hablando el jugador sentado a la izquierda del que ha puesto la ciega grande. Tiene las opciones de *retirarse*, *ir* (ver la apuesta), *subir*, o *all-in* (apostar todo). Si decide *ir* deberá igualar la apuesta mínima (equivalente a la ciega grande). En cambio, si opta por *subir*, deberá superar la apuesta mínima.

Si el primer jugador ha subido, el siguiente jugador puede ir (igualar la máxima apuesta), retirarse, subir, o all-in.

Pasa el turno al siguiente jugador en el sentido de las agujas del reloj, concluyendo las apuestas cuando todos los jugadores que no se han retirado han contribuido con la misma cantidad de dinero.

Si se da el caso de que un jugador sube o apuesta todo y ningún otro iguala su apuesta, gana automáticamente la mano y todo el dinero apostado.

Si un jugador echa un *all-in*, otro puede *ir* (apostando todo su dinero) aunque dicha cantidad sea inferior que la del primer jugador. En caso de que



pierda el que ha aportado una cantidad superior al bote, recuperará la diferencia de dinero apostado.

A partir de ahora antes de cada una de las tres fases restantes, se quema (*burn*) la carta situada en la primera posición de la baraja, que queda boca abajo excluida del juego.

Segunda Fase, Flop:

Los jugadores que no se han retirado ven el *Flop*, que son las tres primeras cartas comunitarias. A continuación se apuesta al igual que en la ronda anterior, salvo que los jugadores tienen también la opción de *pasar* (teniendo derecho a ver o subir cualquier apuesta que se haga detrás de él cuando le vuelva a tocar el turno). Empieza hablando el jugador que está a la izquierda del repartidor, y se continúa en sentido de las agujas del reloj.

Tercera Fase, Turn y Cuarta Fase, River:

Se corresponden con la cuarta y la quinta carta comunitaria sucesivamente. Los jugadores que sigan en la mano deberán apostar de manera idéntica que en el *Flop*.

Al finalizar el *River*, los jugadores que no se hayan retirado comparan sus jugadas, llevándose el bote el que tenga una combinación mayor.

En la siguiente mano, el nuevo repartidor pasa a ser el situado a la izquierda del actual.

Entre las distintas estrategias que se pueden encontrar en los jugadores de póker destacan el *loose-pasivo* que apuesta en el mayor número de manos posibles, el *loose-agresivo*, que va a muchas manos de forma agresiva, el *tight-pasivo*, que juega pocas manos de forma pasiva, y el *tight-agresivo*, que juega pocas manos y cada vez que lo hace, de manera agresiva.

1.2 MARCO REGULADOR

Este proyecto no tiene marco regulador ninguno debido a que consiste en desarrollar un software para realizar un estudio estadístico del Póker al que no se le ha aplicado regulación alguna.



1.3 MARCO SOCIOECONÓMICO.

Refs: [9] y [10]

El volumen de negocio del Póker tiene cifras realmente elevadas. El día 5 de Diciembre del 2008 se batió el record Guinness del torneo online más grande conseguido hasta la fecha, con un premio de medio millón de dólares, en el que participaban más de 35.000 jugadores. Ese mes John “The Razor” Phan fue elegido el mejor jugador del año, ganando una suma total de 2.075.323 de dólares, aunque Tom Dwan “durrrr” ganó más dinero, la cifra de 5.370.000 de dólares.

Remontando en la historia, en 1928, existía un garito de Chicago muy popular y lujoso, regentado por Al Capone y sus secuaces. Los excesos de alcohol y drogas que en el propio lugar se consumían, y misma excitación del juego, daban lugar a que los clientes que lo frecuentaban apostaran grandes cantidades de dinero. En la mesa principal del casino, donde se jugaba la partida más importante, se llegó a formar un bote de más de 2 millones de dólares (lo equivalente a 20 millones de dólares hoy en día).

Este bote fue a parar a manos de un andaluz, quien con jugadas no demasiado buenas, fue forzando al resto de jugadores para que abandonaran la mesa. Su identidad a día de hoy es desconocida. Se piensa que era un estrecho colaborador de Capone, aunque hay quienes creen que era un agente federal al que se le permitió ganar tal suma de dinero.

Hoy en día, el Póker online ha incrementado notablemente el número de jugadores a nivel mundial. Los casinos más convencionales, poco atractivos para los jugadores recién iniciados, no obtienen grandes ganancias debido a lo que les supone mantener estas salas. A menudo consiguen más dinero sustituyéndolas por máquinas tragaperras.

Sin embargo el negocio online obtiene más beneficios y son más baratos debido al poco coste de mantenimiento que les supone. Ofrecen beneficios para los jugadores novatos, puesto que les permiten realizar apuestas de coste mucho más bajo. Su software también les facilita información sobre la cantidad de apuestas realizadas, la jugada que tienen en todo momento y les indican cuando les llega su turno para jugar.

El Póker se puede practicar como un negocio, un juego o un deporte mental. Lo que sí está claro es que la suerte no es el factor más relevante para ganar. La elección de la mesa es la decisión más importante que se puede tomar. Cambiarla es una buena opción si es muy difícil ganar (suele suceder cuando en ella se encuentran jugadores inexpertos o irracionales).



2. ALGORITMOS GENÉTICOS

2.1 DESCRIPCIÓN DE LOS ALGORITMOS GENÉTICOS Refs: [3],[4] y [5]

Los Algoritmos Genéticos (AGs) son métodos de adaptación que se emplean principalmente para problemas de búsqueda y optimización, cuya naturaleza es muy compleja o se desconoce por completo. Pertenecen a la rama de Inteligencia Artificial conocida como Computación Evolutiva o Algoritmos Evolutivos, que trata el estudio de técnicas heurísticas basadas en la teoría de la Evolución de Darwin.

Los Algoritmos Genéticos son muy eficaces para encontrar soluciones a aquellos problemas con gran espacio de búsqueda y que carecen de una determinada técnica para hallarlas. Por tanto permiten realizar tareas muy difíciles de llevar a cabo mediante métodos analíticos y maximizar las ganancias de funciones cuyas variables de entrada no son lineales y tienen interdependencias numéricas muy complejas.

Funcionamiento de los Algoritmos Genéticos

El funcionamiento de los Algoritmos Genéticos se basa en el mecanismo que emplea la naturaleza para la evaluación de las especies. Los individuos que tienen una mejor adaptación al entorno que les rodea sobreviven sobre los que no la tienen, lo que conlleva a una mayor probabilidad de reproducción para producir una nueva generación de la especie con los genes de sus predecesores.

De esta forma, las mejores cadenas genéticas se propagarán a las generaciones sucesoras por mediante cruces de los individuos supervivientes. Esto permite lograr la convergencia hacia la solución del problema, o al menos hallar una solución óptima que estará determinada por la población inicial de los individuos, cuya cadena de ADN se inicia aleatoriamente para lograr la mayor diversidad posible.

Para ello se debe codificar a cada individuo como una solución básica del problema en una cadena de ADN (en este caso la estrategia de un jugador), y crear una función *fitness* que permita evaluar hasta que punto cada individuo evoluciona para poder actuar en consecuencia.

El funcionamiento común para cualquier diseño de Algoritmo Genético viene dado por el siguiente pseudocódigo:



Inicio-AG

Inicializar Población (al azar)

Asignar valores aptitud iniciales (uniformes)

Establecer n como número de generaciones del Algoritmo

desde $i = 0$ hasta n

 obtener evaluación de los individuos mediante la función *fitness*

 seleccionar para reproducción

 reproducir

 mutar

 asignar aptitudes (evaluar individuo)

 evaluar nueva población

 verificar condición de terminación

fin-desde

fin-AG

Figura 1

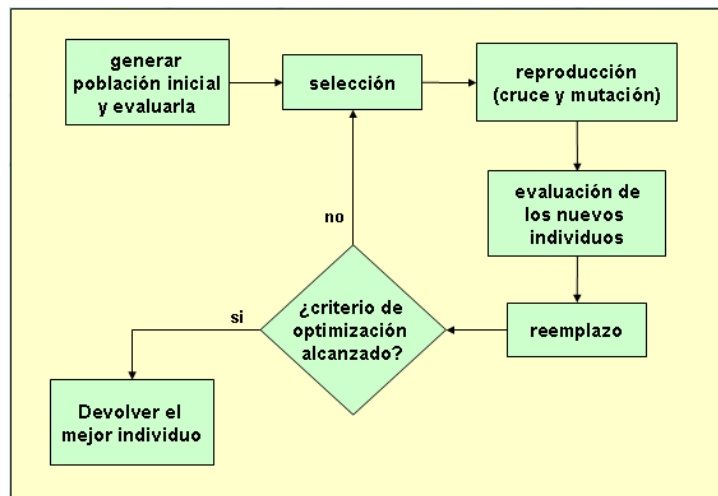


Figura 1: Taller de Algoritmos evolutivos

(<http://atc.ugr.es/pedro/tutoriales/cursos/baeza/ae/documentacion.html>)

Función de evaluación fitness

El buen diseño de la función de evaluación resulta vital para el buen funcionamiento del Algoritmo Genético, ya que representa la mejor o peor adaptación al medio de cada individuo. Para ciertos problemas, la función de evaluación puede suponer un coste computacional demasiado elevado, por lo que se pueden implementar funciones de evaluación aproximada.

Población Inicial

El código genético de la primera población debe ser inicializado aleatoriamente, para conseguir una uniformidad en la creación de los individuos. Existe la posibilidad de seleccionar técnicas heurísticas para la inicialización de los genes, pero los pocos estudios que hay al respecto demuestran que el Algoritmo Genético tiende a converger más rápidamente con el riesgo de producir soluciones máximas locales.

El número de individuos inicial no debe excederse en tamaño para no producir un coste computacional elevado, y no puede ser demasiado pequeño para que el Algoritmo explore todos los espacios de búsqueda. Es aceptable para códigos binarios una cantidad de población comprendida entre la longitud de su cadena genética, y dos veces dicha longitud.



Codificación de los genes

Es uno de los aspectos más importantes para el correcto funcionamiento del Algoritmo Genético. Todas las posibles soluciones deben estar representadas y tienen que ser coherentes al problema, deben tener la misma cantidad de codificaciones (longitud de la cadena).

El mecanismo de codificación de un sólo individuo tiene que ser sencillo, no puede suponer una carga computacional elevada, y pequeños cambios en individuos (mutación) deben suponer pequeños cambios en la solución final.

Selección

Existen varias técnicas de selección de individuos predominantes en una generación que serán cruzados para la creación de nuevos individuos. Una de ellas se denomina función de selección proporcional a la función de evaluación. Como su propio nombre indica, la probabilidad de selección de un individuo será directamente proporcional a su resultado obtenido en la función *fitness*. Utilizando este procedimiento se corre el riesgo de que la población converja prematuramente hacia un máximo local, debido a la aparición de "superindividuos" muy similares entre sí, con baja diversidad genética. Por lo tanto se encontrará una solución buena, pero no la mejor para la población dada.

Otra técnica es la selección Elitista, en la que solamente son candidatos para la reproducción los individuos que tengan una mayor evaluación de *fitness*. También corre el riesgo de producir superindividuos, por lo que no será válida para la resolución de este proyecto.

El método utilizado para hallar una estrategia óptima en el Póker es el que se conoce como método de Torneo, que tiene similitudes con el proceso de selección del mundo animal, donde los machos combaten por el control del grupo. Consiste en hacer competir a los individuos de manera aleatoria, normalmente por parejas (al tratarse de partidas de Póker se utilizará una variante de esta técnica, haciéndolos competir en grupos de nueve jugadores). Este método nos asegurará que al menos dos progenitores del mejor individuo de cada generación actuarán en la siguiente.

Las ventajas de utilizar este último método es que permite comparar a los individuos manteniendo una variabilidad genética, y evitando la aparición de superindividuos.

Reproducción

Una vez seleccionados los individuos que han obtenido mejor resultado en su evaluación, deben ser cruzados (crossover) entre ellos, tal y como hace la naturaleza. El proceso consiste en crear descendientes mediante el intercambio de los genes de sus padres. Existen dos técnicas mostradas en la figura 2 para realizar la reproducción, a partir de un punto de cruce y a partir de dos puntos de cruce:

Figura 2

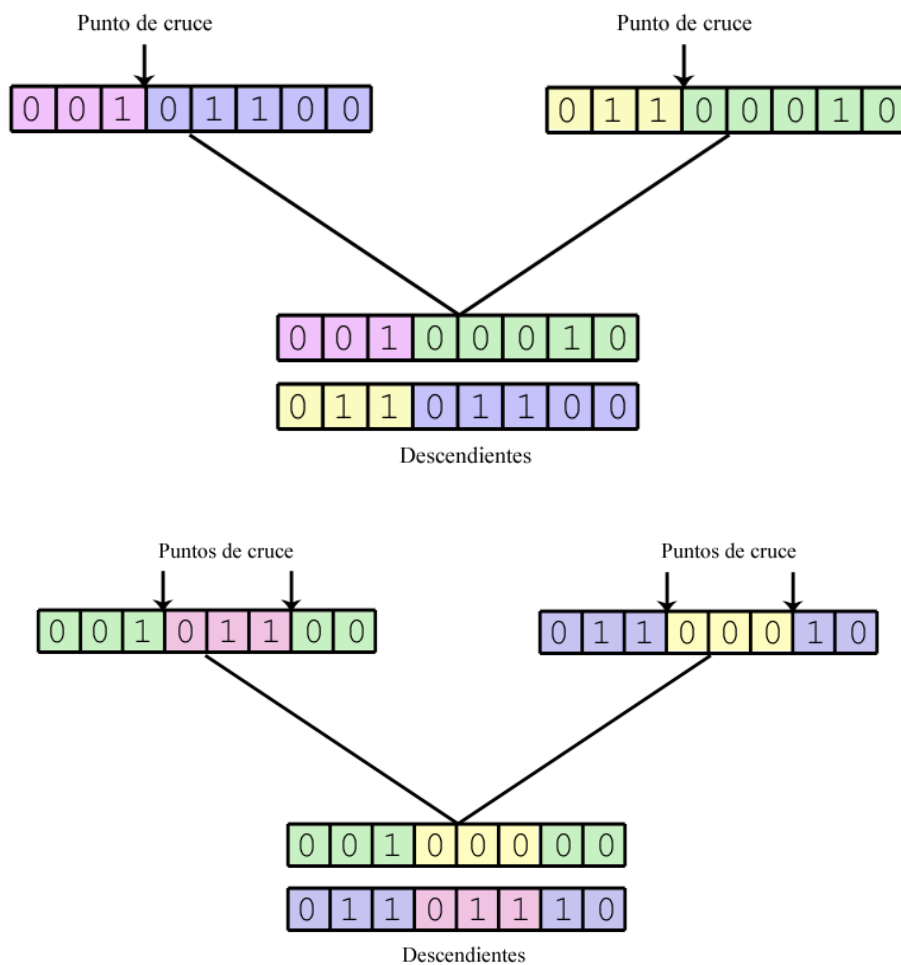


Figura 2: *La Autocondena Humana, Algoritmos Genéticos*
(ramirezsolozano.blogspot.com.es)



Estrategias de reemplazo

Existen dos tipos de Algoritmos Genéticos atendiendo a las estrategias de reemplazo de individuos para formar una nueva generación:

El primer tipo es el Algoritmo Genético Generacional, que reemplaza toda la población por sus predecesores.

El segundo tipo es el Algoritmo Genético de Estado Estacionario que reemplazan N individuos, utilizando uno de los siguientes criterios:

- Los N individuos con peor evaluación son sustituidos por los nuevos.
- Los nuevos individuos reemplazan a sus N progenitores.
- Los nuevos individuos reemplazan a los más parecidos entre sí ya existentes.

Normalmente el método de reemplazo más común es el primero de los mencionados anteriormente, aunque si se utiliza el segundo, los individuos que perduran o sobreviven de una generación a otra representan un porcentaje muy pequeño de la población.

Mutación

Tal y como ocurre en las cadenas de ADN de los seres vivos, en los Algoritmos Genéticos se aplica durante el proceso de cruce el operador de mutación. Se modifican uno o varios genes de los individuos descendientes de la anterior generación. Esta técnica permite aumentar la diversidad genética favoreciendo la aparición de nuevos individuos, posibles soluciones al problema que se intenta solucionar.

También ayuda a evitar máximos locales cuando han transcurrido un cierto número de poblaciones. La probabilidad de mutación debe ser baja (con valores habituales menores del 1% o 0.1%), dado que si se abusa de este procedimiento estaremos empleando estos algoritmos como una técnica de búsqueda al azar.

Figura 3

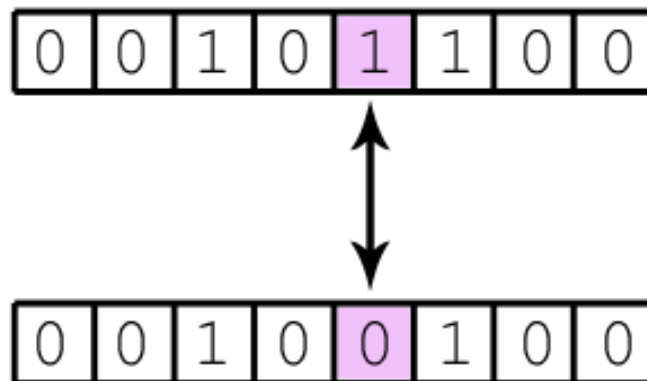


Figura 3: *La Autocondena Humana, Algoritmos Genéticos*
(ramirezsolozano.blogspot.com.es)

Convergencia

La implementación de un Algoritmo Genético se utiliza para hallar la solución un problema determinado, el cual no tiene otra técnica específica para resolverlo. Esto implica que no podemos saber si el algoritmo ha encontrado la mejor solución posible o una solución aproximada (máximo local), a no ser que para un caso concreto sepamos a priori cual es la mejor solución.

A día de hoy no existe ningún método que asegure la correcta convergencia de un Algoritmo Genético. Dependiendo de cada problema, se pueden utilizar técnicas que pueden resultar útiles.

Una de ellas consiste en ejecutar el algoritmo durante un determinado número de generaciones, detenerlo y emplear el mejor cromosoma de los individuos de la última generación como resultado al problema. Esta técnica, muy utilizada en inteligencia artificial, es muy útil cuando el tiempo de proceso del algoritmo es crítico (algoritmos en tiempo real) y nos vale con la mejor solución encontrada aunque no sea la óptima.

Otro método muy utilizado es procesar el algoritmo hasta que la mayoría de la población converja. Esto sucederá cuando gran parte de los individuos tenga un código genético muy similar y una adaptación al medio (valor de *fitness*) muy próxima.

Lo ideal sería que un Algoritmo Genético convergiera rápidamente hacia la solución óptima del problema, aunque normalmente lo normal es que se quede atrapado parcial o indefinidamente en un óptimo local.

No hay ninguna técnica que dirija al algoritmo hacia la mejor solución, pero se pueden emplear métodos para minimizar los efectos de estancamientos en máximos locales, estableciendo correctamente los operadores de selección, cruce y mutación, que serán distintos dependiendo del problema a tratar.

Se pueden implementar Algoritmos Genéticos en paralelo (conocido como el Modelo de Islas). Consiste en crear dos o más subpoblaciones, ejecutando en cada una de ellas un Algoritmo Genético, y cada cierto tiempo realizar migraciones de una subpoblación a otra para conseguir un intercambio de información. Esto reducirá la probabilidad de estancamientos en máximos locales, aumentará la diversidad genética (la solución del algoritmo no sólo depende de la inicialización de una primera población), y guiará el algoritmo hacia una mejor solución. Para ello se proponen distintas comunicaciones en la figura 4.

Figura 4

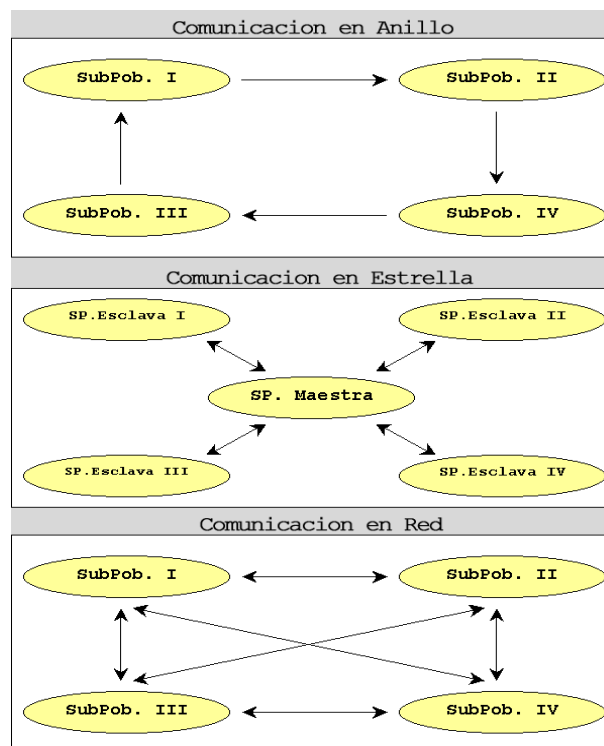


Figura 4: *Algoritmos Genéticos*, Javier Ventoso Reigosa (Artículo publicado en la revista Sólo Programadores Num. 89, pg 8).



Ventajas y Limitaciones

Una de las ventajas más importantes de los Algoritmos Genéticos es que trabajan simultáneamente con varias soluciones y no necesita conocer el problema para hallar el resultado final. Será la función de evaluación *fitness* la que se encargue de guiar el desarrollo algoritmo hacia una mejor solución. Su fácil implementación en paralelo aumenta el espacio de búsqueda y evita estancamientos locales. Son realmente buenos para la exploración de problemas con un espacio de búsqueda enorme y para los que no existe un método determinista para solucionarlos, como se da en el caso de una estrategia de Póker.

La estrategia de búsqueda de los Algoritmos Genéticos incrementa exponencialmente a los individuos con un mejor valor de *fitness*, pero sin dejar de probar nuevas soluciones.

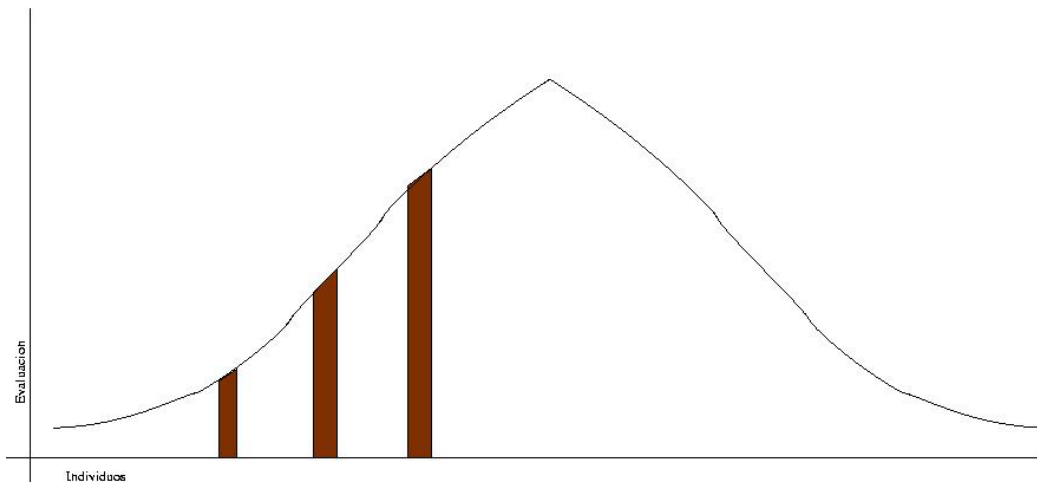
En cambio, tienen limitaciones que pueden producir que un Algoritmo Genético no sea el mejor método para resolver determinados casos. Pese a que el diseño de un algoritmo sea adecuado, no tenemos garantías de que la solución encontrada sea un máximo global, corremos el riesgo de que se quede estancado localmente y tarde mucho en salir, o no salga nunca.

Tienen gran dependencia de los operadores de selección, cruce, mutación que seleccionemos, así como de su implementación en paralelo (migración de población), del diseño de la función *fitness*, del tamaño de la población inicial, condición de parada, codificación de la cadena de ADN de los individuos, número de generaciones a evaluar, etc.

También el óptimo encontrado por el algoritmo (ya sea local o global) depende del código genético elegido aleatoriamente de la población inicial. Con el paso de las generaciones se pierde variabilidad genética, ya que los individuos son cada vez más parecidos entre sí para lograr la convergencia.

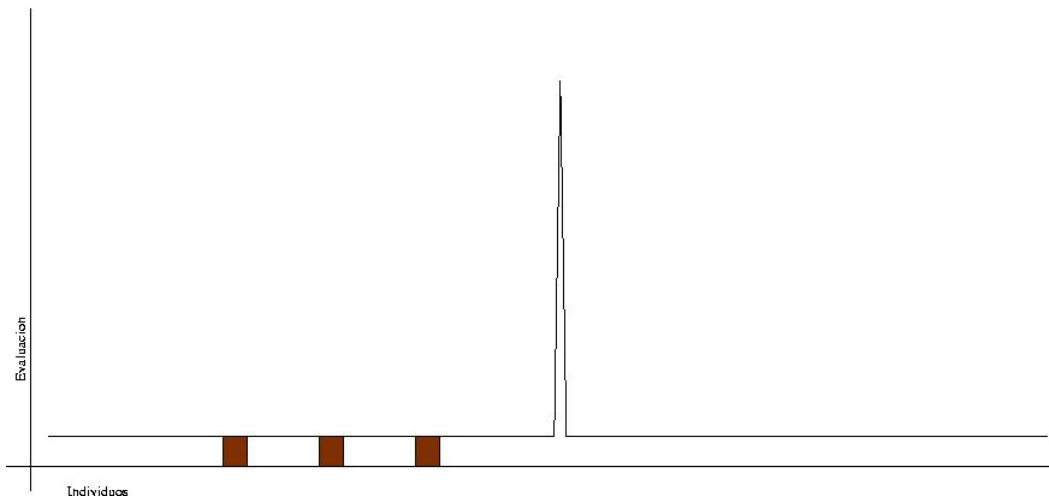
Para poder encontrar una solución, la población, antes de alcanzar su convergencia, debe tener valores de *fitness* suficientemente diferentes:

Figura 5



En caso contrario, se dice que la evaluación de la función es demasiado abrupta:

Figura 6



Figuras: *Algoritmos Genéticos*, Pedro Isasi (Asignatura Computación Biológica Departamento de Informática Universidad Carlos III de Madrid, pg 112)

Aplicaciones de los Algoritmos Genéticos

Hoy en día se utilizan en ingeniería aplicados a una amplitud de ramas, como por ejemplo: diseño automatizado de equipación industrial, diseño automatizado de sistemas de comercio en el sector financiero, diseño de sistemas de distribución de aguas, diseño de topologías de circuitos impresos y de redes computacionales, predicción, diseño de horarios de grandes universidades para evitar conflictos de solapamiento de clases, hallazgo de errores en programas, y Teoría de los juegos (como se trata del Póker)...



2.2 TEORÍA DE LOS JUEGOS

Refs: [6],[7] y [8]

La Teoría de los Juegos constituye una extensión de la Teoría de la Decisión, que estudia la toma de decisiones por parte de individuos cuando existe una incertidumbre, que puede ser objetiva o subjetiva, que viene dada por las acciones de otros decisores.

La Teoría de los Juegos no está pensada para juegos de azar, la relación que tiene con este trabajo es el concepto de “estrategia mixta”, que a diferencia de una “estrategia pura” (que responde de manera determinista y unívoca ante una determinada situación), puede responder de varias formas, de acuerdo con una distribución de probabilidades, que es precisamente la forma en la que hemos definido nuestro problema.

Un juego consiste en un conjunto de individuos dotados de una estrategia o movimientos que determinarán sus acciones y una especificación de recompensas para cada una de las combinaciones de las estrategias. En este caso la recompensa será el dinero ganado por cada jugador en una partida de Póker.

La estrategia óptima de la teoría de los juegos es la que consigue mayor evaluación como salida de la función *fitness* cuando un oponente elige la mejor contraestrategia posible, solución que se trata de hallar aplicando los Algoritmos Genéticos.

Entendiendo como funcionan los distintos escenarios posibles, se pueden detectar desviaciones de las estrategias de los oponentes respecto a la óptima de la teoría de los juegos pudiendo maximizar el margen sobre dichas estrategias y explotarlas.

2.3 EL CRITERIO DE KELLY

Refs: [11]

Se conoce como el criterio de *Kelly*, en la teoría de las probabilidades, a la fórmula que determina el porcentaje óptimo de "bankroll", que se debe emplear en una determinada apuesta, para maximizar el crecimiento del mismo a largo plazo.

Fue desarrollado por Jonh *Kelly* en 1956. Aunque en un principio fue pensado para apuestas de carreras de caballos, es válido para cualquier otro tipo de apuesta.



La fórmula es la siguiente:

$$\text{Bankroll}\% = \frac{C * \frac{P}{100} - 1}{C - 1} * 100$$

Donde:

- Bankroll es la cantidad de dinero que tienes para apostar.
- C es la cuota que te ofrece la casa de apuestas (en el caso del Póker el bote total en mesa).
- P es la estimación de la probabilidad de ganar la apuesta estimada en tanto por ciento.

Problemas del criterio de Kelly

La estimación de la probabilidad puede ser subjetiva, aunque hay personas que utilizan métodos más científicos. Si aseguramos que nuestro estimador de probabilidad es el mejor posible, entonces el criterio de *Kelly* nos garantiza beneficio seguro a largo plazo.

El porcentaje de bankroll obtenido mediante este criterio puede suponer un riesgo demasiado elevado, sobre todo si se estima mal la probabilidad y si nos encontramos ante una mala racha de apuestas.

2.4 PROBABILIDADES EN EL PÓKER, APLICACIÓN DEL CRITERIO DE KELLY

En este apartado se trata de explicar cómo se calcula la cuota y la probabilidad para obtener el porcentaje de bankroll del criterio de *Kelly*.

En el Póker no se indica la cuota, pero es fácil de calcular como:

$$\text{Cuota } (C) = \frac{\text{Cantidad de dinero o fichas totales en mesa}}{\text{Apuesta individual}}$$

Es decir, si por ejemplo en una mesa hay 600 dólares y un jugador contribuye con 100 para subir la apuesta, su cuota calculada sería de 6.



La probabilidad de ganar la apuesta es más fácil de estimar que en un partido de fútbol o una carrera de caballos, ya que aquí no existe el nivel de subjetividad en cuanto a qué equipo es mejor y en qué porcentaje supera a su rival.

Para realizar este cálculo es necesario recurrir a la información de las probabilidades de que salga cada tipo de jugada:

$$\text{Probabilidad Escalera Real} = \frac{4}{2598960} = 0.0001 \%$$

$$\text{Probabilidad Escalera Color} = \frac{36}{2598960} = 0.001 \%$$

$$\text{Probabilidad Póker} = \frac{624}{2598960} = 0.024 \%$$

$$\text{Probabilidad Full} = \frac{3774}{2598960} = 0.14 \%$$

$$\text{Probabilidad Color} = \frac{5108}{2598960} = 0.19 \%$$

$$\text{Probabilidad Escalera} = \frac{10200}{2598960} = 0.39 \%$$

$$\text{Probabilidad Trío} = \frac{54912}{2598960} = 2.11 \%$$

$$\text{Probabilidad Dobles Parejas} = \frac{123552}{2598960} = 4.7 \%$$

$$\text{Probabilidad Pareja} = \frac{1098240}{2598960} = 42 \%$$

$$\text{Ninguna jugada} = \frac{1302540}{2598960} = 50.1 \%$$

Datos extraídos de <http://www.poker.com.mx/cartas/estadisticas-jugadas-poker.html>

Estos datos anteriores se calculan para cada caso como número total de jugadas que existen de cada tipo dividido entre el número total de combinaciones de cartas que existen en el Póker.

Cabe destacar la rápida disminución de probabilidad de conseguir una jugada a medida que ésta aumenta en valor.



Conociendo los datos anteriores, se calcula la probabilidad aproximada de ganar con una cierta jugada como la suma de probabilidades de que el resto de jugadores activos (que no estén eliminados) tengan jugadas inferiores. Por ejemplo:

$$P. \text{Ganar con Trío} = (P. \text{Doble Pareja} + P. \text{Pareja} + P. \text{Ninguna Jugada})^n$$

Donde n es el número de jugadores activos.

Si se quiere calcular la probabilidad de ganar con una jugada que no se ha conseguido todavía (por ejemplo un jugador tiene cartas propensas a formar escalera), se debe multiplicar por la probabilidad de conseguir dicha jugada como muestra el siguiente ejemplo:

$$P. \text{Ganar Escalera} =$$

$$= P. \text{Escalera} * (P. \text{Trio} + P. \text{Dobles Parejas} + P. \text{Parejas} + P. \text{Ninguna Jugada})^n$$

Cuatro funciones de MATLAB se encargan de realizar dichos cálculos dependiendo de la jugada que tenga cada individuo.

Es importante decir que la estimación de probabilidades en el Póker siguiendo este método es una aproximación, debido a que las probabilidades que tiene un individuo de ganar también depende de las cartas que hay en la mesa, y es influida por las probabilidades de ganar de los demás jugadores que también dependen de las cartas que hay en el tapete.

El criterio de *Kelly* aplicado al Póker dará resultados absurdos en muchas ocasiones, ya que la cuota suele ser muy elevada y, para casi todas las combinaciones de alto valor (superiores a dobles parejas o tríos), las probabilidades de conseguirlas son muy bajas.



3. DISEÑO DE LA SOLUCIÓN TÉCNICA

3.1 DISEÑO DE LA ESTRATEGIA

El primer paso a seguir para la resolución del problema que se plantea es el diseño de la estrategia que identifica a un jugador de Póker, cuya estructura es común para todos los individuos. La estrategia juega el papel de cadena genética de la población en el Algoritmo Genético; su evaluación como salida de la función *fitness* determinará la calidad de cada jugador.

La estrategia debe ser planteada cuidadosamente, ya que de ello depende el diseño de la función de evaluación y la calidad de las soluciones obtenidas por el programa. Tiene que definir con el máximo detalle posible el comportamiento de un jugador de Póker, teniendo en cuenta todas las posibles acciones que puede realizar en distintas situaciones, sin excederse en tamaño o longitud genética, ya que de ser así aumentaría demasiado el coste computacional y es posible que los resultados obtenidos sean dispares.

Está representada por un array de 160 genes, cada uno de ellos es un número del 0 al 100 que determina la probabilidad de tomar decisiones importantes en el juego. La estructura de la estrategia viene detallada a continuación:

Gen Ronda/ Descripción

Prefloop

- 1: Probabilidad de Farol. Determina la probabilidad de un individuo de subir una apuesta (proporcional a la agresividad) cuando no tiene cartas buenas para hacerlo.
- 2: Porcentaje de agresividad del jugador. Cuanto más alta sea más dinero apostará al realizar una subida.
- 3: Porcentaje de actividad del jugador. Cuanto más alta sea menos probabilidades tendrá de pasar o retirarse en una jugada.
- 4: Probabilidad de que un jugador apueste, suba o vea una subida siguiendo el criterio de *Kelly*.
- 5: Probabilidad de subir/ ver subida teniendo carta alta (figura o as).
- 6: Probabilidad de subir/ ver subida teniendo pares.
- 7: Probabilidad de subir/ ver subida teniendo pares altos.



- 8: Probabilidad de subir/ ver subida teniendo dos cartas propensas a formar escalera.
- 9: Probabilidad de subir/ ver subida teniendo dos cartas del mismo palo (propensas a formar color).
- 10: Probabilidad de subir/ ver subida teniendo dos cartas propensas a formar escalera de color.
- 11: Probabilidad de subir/ ver subida teniendo dos cartas propensas a formar escalera real.
- 12: Probabilidad de echar/ ver all-in teniendo carta alta.
- 13: Probabilidad de echar/ ver all-in teniendo pares.
- 14: Probabilidad de echar/ ver all-in teniendo pares altos.
- 15: Probabilidad de echar/ ver all-in teniendo dos cartas propensas a formar escalera.
- 16: Probabilidad de echar/ ver all-in teniendo dos cartas del mismo palo.
- 17: Probabilidad de echar/ ver all-in teniendo dos cartas propensas a formar escalera de color.
- 18: Probabilidad de echar/ ver all-in teniendo dos cartas propensas a formar escalera real.
- 19: Probabilidad de ver una subida si la ha visto más del 75% de los jugadores.
- 20: Probabilidad de ver una subida si la han visto entre el 50% y el 75% de los jugadores.
- 21: Probabilidad de ver una subida si la han visto entre el 25% y el 50% de los jugadores.
- 22: Probabilidad de ver una subida si la han visto entre el 0% y el 25% de los jugadores.
- 23: Probabilidad de ver allin si el jugador que lo ha echado tiene más dinero que el individuo.
- 24: Probabilidad de ver allin si el jugador que lo ha echado tiene menos dinero que el individuo.
- 25: Probabilidad de ver allin si lo han visto más individuos.
- 26: Probabilidad de retirarse si el jugador tiene poco dinero.



Flop

- 27: Probabilidad de farol.
- 28: Porcentaje de agresividad.
- 29: Porcentaje de actividad.
- 30: Probabilidad de seguir el criterio de *Kelly*.
- 31: Probabilidad de subir/ver subida teniendo carta alta.
- 32: Probabilidad de subir/ver subida teniendo pareja.
- 33: Probabilidad de subir/ver subida teniendo pareja alta.
- 34: Probabilidad de subir/ver subida teniendo dobles parejas.
- 35: Probabilidad de subir/ver subida teniendo dobles (alta).
- 36: Probabilidad de subir/ver subida teniendo trio.
- 37: Probabilidad de subir/ver subida teniendo trio alto.
- 38: P. de subir/ver subida teniendo 3 cartas propensas a formar escalera.
- 39: P. de subir/ver subida teniendo 4 cartas propensas a formar escalera.
- 40: Probabilidad de subir/ver subida teniendo escalera.
- 41: Probabilidad de subir/ver subida teniendo 3 cartas del mismo color.
- 42: Probabilidad de subir/ver subida teniendo 4 cartas del mismo color.
- 43: Probabilidad de subir/ver subida teniendo color.
- 44: Probabilidad de subir/ver subida teniendo full.
- 45: Probabilidad de subir/ver subida teniendo póker.
- 46: P.de subir/ver subida teniendo 3 cartas propensas a formar esc de color.
- 47: P.de subir/ver subida teniendo 4 cartas propensas a formar esc de color.
- 48: Probabilidad de subir/ver subida teniendo escalera de color.
- 49: P. de subir/ver subida teniendo 3 cartas propensas a formar esc real.
- 50: P. de subir/ver subida teniendo 4 cartas propensas a formar esc real.
- 51: Probabilidad de subir/ver subida teniendo escalera real.
- 52: Probabilidad de echar/ver all-in teniendo carta alta.
- 53: Probabilidad de echar/ver all-in teniendo pareja.



- 54: Probabilidad de echar/ver all-in teniendo pareja alta.
- 55: Probabilidad de echar/ver all-in teniendo dobles parejas.
- 56: Probabilidad de echar/ver all-in teniendo dobles (alta).
- 57: Probabilidad de echar/ver all-in teniendo trio.
- 58: Probabilidad de echar/ver all-in teniendo trio alto.
- 59: Probabilidad de echar/ver all-in teniendo 3 cartas propensas a formar escalera.
- 60: P. de echar/ver all-in teniendo 4 cartas propensas a formar escalera.
- 61: Probabilidad de echar/ver all-in teniendo escalera.
- 62: Probabilidad de echar/ver all-in teniendo 3 cartas del mismo color.
- 63: Probabilidad de echar/ver all-in teniendo 4 cartas del mismo color.
- 64: Probabilidad de echar/ver all-in teniendo color.
- 65: Probabilidad de echar/ver all-in teniendo full.
- 66: Probabilidad de echar/ver all-in teniendo póker.
- 67: P. de echar/ver all-in teniendo 3 cartas propensas a formar esc de color.
- 68: P. de echar/ver all-in teniendo 4 cartas propensas a formar esc de color.
- 69: Probabilidad de echar/ver all-in teniendo escalera de color.
- 70: P. de echar/ver all-in teniendo 3 cartas propensas a formar esc real.
- 71: P. de echar/ver all-in teniendo 4 cartas propensas a formar escalera real.
- 72: Probabilidad de echar/ver all-in teniendo escalera real.
- 73: P.de ver una subida si la ha visto entre el 75% y 100% de los jugadores.
- 74: P.de ver una subida si la ha visto entre el 50% y 75% de los jugadores.
- 75: P.de ver una subida si la ha visto entre el 25% y 50% de los jugadores.
- 76: P. de ver una sibida si la ha visto el 0-25% de los jugadores.
- 77: Probabilidad de ver un all-in si el jugador que lo ha echado tiene más dinero que el individuo.
- 78: Probabilidad de ver un all-in si el jugador que lo ha echado tiene menos dinero que el individuo.
- 79: Probabilidad de ver un all-in si lo han visto más jugadores.



Turn

- 80: Probabilidad de farol.
- 81: Porcentaje de agresividad.
- 82: Porcentaje de actividad.
- 83: Probabilidad de seguir el criterio de *Kelly*.
- 85: Probabilidad de subir/ver subida teniendo pareja.
- 86: Probabilidad de subir/ver subida teniendo pareja alta.
- 87: Probabilidad de subir/ver subida teniendo dobles parejas.
- 88: Probabilidad de subir/ver subida teniendo dobles parejas alta.
- 89: Probabilidad de subir/ver subida teniendo trío.
- 90: Probabilidad de subir/ver subida teniendo trío alto.
- 91: P. de subir/ver subida teniendo 4 cartas propensas a formar escalera.
- 92: Probabilidad de subir/ver subida teniendo escalera.
- 93: P. de subir/ver subida teniendo 4 cartas del mismo color.
- 94: Probabilidad de subir/ver subida teniendo color.
- 95: Probabilidad de subir/ver subida teniendo full.
- 96: Probabilidad de subir/ver subida teniendo póker.
- 97: P.de subir/ver subida teniendo 4 cartas propensas a formar esc de color.
- 98: Probabilidad de subir/ver subida teniendo escalera de color.
- 99: P.de subir/ver subida teniendo 4 cartas propensas a formar esc real.
- 100: Probabilidad de subir/ver subida teniendo escalera real.
- 101: Probabilidad de echar/ver all-in teniendo carta alta.
- 102: Probabilidad de echar/ver all-in teniendo pareja.
- 103: Probabilidad de echar/ver all-in teniendo pareja alta.
- 104: Probabilidad de echar/ver all-in teniendo dobles parejas.
- 105: Probabilidad de echar/ver all-in teniendo dobles parejas alta.
- 106: Probabilidad de echar/ver all-in teniendo trío.
- 107: Probabilidad de echar/ver all-in teniendo trío alto.



- 108: P.de echar/ver all-in teniendo 4 cartas propensas a formar escalera.
- 109: Probabilidad de echar/ver all-in teniendo escalera.
- 110: Probabilidad de echar/ver all-in teniendo 4 cartas del mismo color.
- 111: Probabilidad de echar/ver all-in teniendo color.
- 112: Probabilidad de echar/ver all-in teniendo full.
- 113: Probabilidad de echar/ver all-in teniendo póker.
- 114: P. de echar/ver all-in teniendo 4 cartas propensas a formar esc de color.
- 115: Probabilidad de echar/ver all-in teniendo escalera de color.
- 116: P de echar/ver all-in teniendo 4 cartas propensas a formar escalera real.
- 117: Probabilidad de echar/ver all-in teniendo escalera real.
- 118: P. de ver una subida si la ha visto entre el 75% - 100% de los jugadores.
- 119: P. de ver una subida si la ha visto entre el 50% - 75% de los jugadores.
- 120: P. de ver una subida si la ha visto entre el 25% - 50% de los jugadores.
- 121: P. de ver una subida si la ha visto entre el 0-25% de los jugadores.
- 122: Probabilidad de ver un all-in si el jugador que lo ha echado tiene más dinero que el individuo.
- 123: Probabilidad de ver un all-in si el jugador que lo ha echado tiene menos dinero que el individuo.
- 124: Probabilidad de ver un all-in si lo han visto más jugadores.

River

- 125: Probabilidad de que el individuo sea farolero.
- 126: Porcentaje agresividad.
- 127: Porcentaje actividad.
- 128: Probabilidad apostar por encima del criterio de *Kelly*.
- 129: Probabilidad de subir/ver subida teniendo carta alta.
- 130: Probabilidad de subir/ver subida teniendo pareja.
- 131: Probabilidad de subir/ver subida teniendo pareja alta.
- 132: Probabilidad de subir/ver subida teniendo dobles parejas.
- 133: Probabilidad de subir/ver subida teniendo dobles parejas altas.



- 134: Probabilidad de subir/ver subida teniendo trío.
- 135: Probabilidad de subir/ver subida teniendo trío alto.
- 136: Probabilidad de subir/ver subida teniendo escalera.
- 137: Probabilidad de subir/ver subida teniendo color.
- 138: Probabilidad de subir/ver subida teniendo full.
- 139: Probabilidad de subir/ver subida teniendo póker.
- 140: Probabilidad de subir/ver subida teniendo escalera de color.
- 141: Probabilidad de ver/echar all-in teniendo carta alta.
- 142: Probabilidad de ver/echar all-in teniendo pareja.
- 143: Probabilidad de ver/echar all-in teniendo pareja alta.
- 144: Probabilidad de ver/echar all-in teniendo dobles parejas.
- 145: Probabilidad de ver/echar all-in teniendo dobles parejas altas.
- 146: Probabilidad de ver/echar all-in teniendo trío.
- 147: Probabilidad de ver/echar all-in teniendo trío alto.
- 148: Probabilidad de ver/echar all-in teniendo escalera.
- 149: Probabilidad de ver/echar all-in teniendo color.
- 150: Probabilidad de ver/echar all-in teniendo full.
- 151: Probabilidad de ver/echar all-in teniendo póker.
- 152: Probabilidad de ver/echar all-in teniendo escalera de color.
- 153: Probabilidad de ver/echar all-in teniendo escalera real.
- 154: P. de ver una subida si la ha visto entre el 75% -100% de los jugadores.
- 155: P. de ver una subida si la ha visto entre el 50% -75% de los jugadores.
- 156: P. de ver una subida si la ha visto entre el 25% -50% de los jugadores.
- 157: P. de ver una subida si la ha visto entre el 0 -25% de los jugadores.
- 158: Probabilidad de ver un all-in si el jugador que lo ha echado tiene más dinero que el individuo.
- 159: Probabilidad de ver un all-in si el jugador que lo ha echado tiene menos dinero que el individuo.
- 160: Probabilidad de ver un all-in si lo han visto más jugadores.

Inicialización de la estrategia

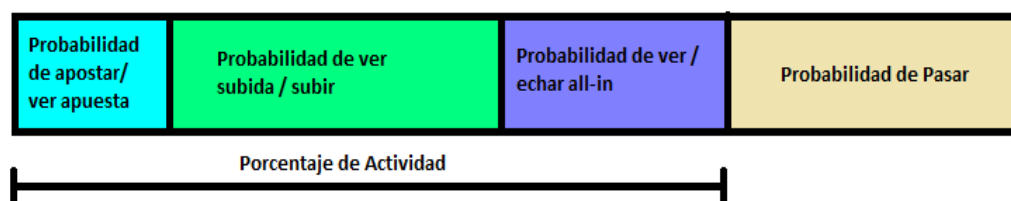
Como se ha explicado con anterioridad en el apartado de Algoritmos Genéticos, todos los genes se deben inicializar aleatoriamente para conseguir una distribución uniforme para los individuos y evitar una convergencia prematura de los mismos, que puede producir máximos locales, en la ejecución del algoritmo.

Los únicos genes inicializados con un valor fijo (100%) son los definidos como probabilidad de ver/ echar all-in teniendo escalera real (en las rondas de Flop, Turn y River), ya que es la mayor combinación de cartas posible en este juego.

Para establecer las probabilidades de subir/ ver subida y echar/ ver all-in para cada combinación de cartas en las cuatro rondas de apuestas, primero se inicializa el porcentaje de actividad como un número aleatorio entre 0 y 100. En función de esta cifra, se crean las otras dos probabilidades de manera que la suma de apostar, subir y all-in sea igual al porcentaje de actividad, y la suma de este último con su opuesto (probabilidad de pasar) sea igual a 100%.

Las probabilidades de pasar y apostar no están definidas en la estrategia. Estos datos se extraen de los porcentajes antes mencionados, como se muestra en la figura 7.

Figura 7



3.2 CODIFICACIÓN DE LA BARAJA DE CARTAS

La baraja utilizada en este juego es de tipo inglés de 52 cartas. Tiene cuatro palos (corazones, picas, diamantes y tréboles) que son del mismo valor, ya que ninguno prevalece sobre otro. Por tanto no influyen en el orden de su codificación. Los naipes están representados en memoria con números del 1 al 52 (13 de cada palo) como se muestra en la figura 8.

Figura 8

	As	2	3	4	5	6	7	8	9	10	J	Q	K
Corazones	1	2	3	4	5	6	7	8	9	10	11	12	13
Picas	14	15	16	17	18	19	20	21	22	23	24	25	26
Diamantes	27	28	29	30	31	32	33	34	35	36	37	38	39
Tréboles	40	41	42	43	44	45	46	47	48	49	50	51	52

3.3 IDENTIFICACIÓN DE LA JUGADA

Gracias a la codificación de las cartas propuesta anteriormente es fácil identificar las distintas jugadas que tienen los individuos.

Por ejemplo, para detectar pares se buscan cartas que resten 13 o múltiplos de 13. Para las dobles parejas, tríos, full y póker se realiza de manera similar, ya que al igual que los pares son combinaciones de cartas iguales.

Para identificar si un jugador tiene color es muy sencillo; basta con comprobar que cinco de sus cartas están comprendidas entre el rango de valores al que pertenecen los naipes de cada palo.

El algoritmo para identificar escaleras es un poco más complejo: se detectan cartas que sean próximas en valor. Si se llegan a detectar cinco, hay escalera. Para la escalera de color y la escalera real real el algoritmo es igual, con la salvedad de que hay que tener en cuenta el tipo de escalera que son por definición (escalera con cartas del mismo palo, y escalera con cartas del mismo palo 10-As).



Se encargan varias funciones programadas en MATLAB de detectar las distintas combinaciones como se explica en la página anterior. Estas funciones devuelven cuatro parámetros que describen con más detalle como es la jugada:

- **Primer parámetro:** Número de cartas que forman la jugada. Ejemplo: 2 para pares, 3 para tríos, 5 para full, color y escaleras, 3 y 4 para posibles escaleras y color, etc.
- **Segundo parámetro:** Número de cartas de dicha jugada que tiene el jugador en la mano. Su valor es 0, 1 o 2. Se utiliza para comprobar que cada combinación de cartas está formada al menos por una de las individuales y el resto de las comunes.
- **Tercer parámetro:** Valor de la carta más alta perteneciente a la jugada. Se utiliza para desempates.
- **Cuarto parámetro:** Valor de la segunda carta más alta perteneciente a la jugada. Se utiliza para los casos en los que el tercer parámetro es común para dos jugadores. Por ejemplo, hay un empate de dobles parejas. Un individuo tiene (8 8 5 5) y otro (8 8 7 7). El tercer parámetro para ambos sería el 8, pero el cuarto parámetro es mayor para el segundo jugador (7) que para el primero (5). Por tanto en este caso ganaría el segundo.

El siguiente paso es reconocer las distintas jugadas que tiene el individuo para que la función fitness, encargada de simular la partida de Póker, sea capaz de determinar quién es el ganador en cada mano (se explicará con más detalle en el apartado: *Diseño de la función fitness*).

Una forma muy sencilla de guardar en memoria estos datos es en una matriz de diez filas y cuatro columnas, en la que cada fila representa el tipo de jugada en orden descendente (escalera real, color, póker, full, color, escalera, trío, dobles parejas, pareja y carta alta); y cada columna representa los cuatro parámetros explicados anteriormente.

La mejor forma de entenderlo es con los ejemplos que se proponen en la figura 9 y 10.



Ejemplo 1:

Figura 9

1º Parámetro	2º Parámetro	3º Parámetro	4º Parámetro	
Cartas Jugada	Cartas Mano	Carta Alta	Carta Alta 2	
0	0	0	0	0 Escalera Real
3	1	9	7	7 Escalera Color
0	0	0	0	0 Póker
0	0	0	0	0 Full
4	2	9	7	7 Color
5	2	10	9	9 Escalera
0	0	0	0	0 Trío
0	0	0	0	0 Dobles Parejas
0	0	0	0	0 Pareja
1	1	10	9	9 Carta Alta

Este primer individuo tiene tres cartas propensas a formar escalera de color (una de ellas individuales, las otras dos comunes), siendo la más alta un 9 y la segunda más alta un 7.

Tiene cuatro cartas propensas a formar color (dos de ellas individuales, las otras dos comunes), siendo la más alta un 9 y la segunda más alta un 7.

Tiene cinco cartas que forman una escalera (dos de ellas individuales, las otras tres comunes), siendo la carta más alta un 10 y la segunda más alta un 9. Ya que es la primera jugada completa más alta por este jugador, es la que se tendrá en cuenta para comparar con el resto de individuos para obtener un ganador. Es una escalera de 10, 9, 8, 7 y 6.

En este caso la fila que identifica la carta más alta es inservible porque ya disponemos de esta información, pero puede ser de gran utilidad para resolver otro tipo de empates en parejas o tríos por ejemplo.



Ejemplo 2:

Figura 10

1º Parámetro	2º Parámetro	3º Parámetro	4º Parámetro	
Cartas Jugada	Cartas Mano	Carta Alta	Carta Alta 2	
0	0	0	0	0 Escalera Real
0	0	0	0	0 Escalera Color
0	0	0	0	0 Póker
5	1	11	6	6 Full
0	0	0	0	0 Color
2	1	13	11	11 Escalera
3	1	6	0	0 Trío
4	1	11	6	6 Dobles Parejas
2	0	11	0	0 Pareja
1	1	13	0	0 Carta Alta

Este ejemplo muestra como un individuo tiene full como jugada más alta (con una carta individual). En este caso es un full de (11 11 6 6 6). Esta información se obtiene en la fila del trío, que indica que tiene tres 6s con una carta individual.

Para comparar a este jugador con los demás, como en todos los casos, hay que mirar la primera jugada completa más alta (full). Los valores de trío, dobles parejas y pareja vienen implícitos en el full, ya que está compuesto por estas jugadas más bajas.

En este ejemplo el valor de la carta más alta es de utilidad, ya que de haber empate de full con otro jugador se recurre a esta cifra, que en este caso es un Rey (K).

De estos datos se pueden extraer otras conclusiones como la de que este jugador tiene en la mano un Rey (K) y un 6.

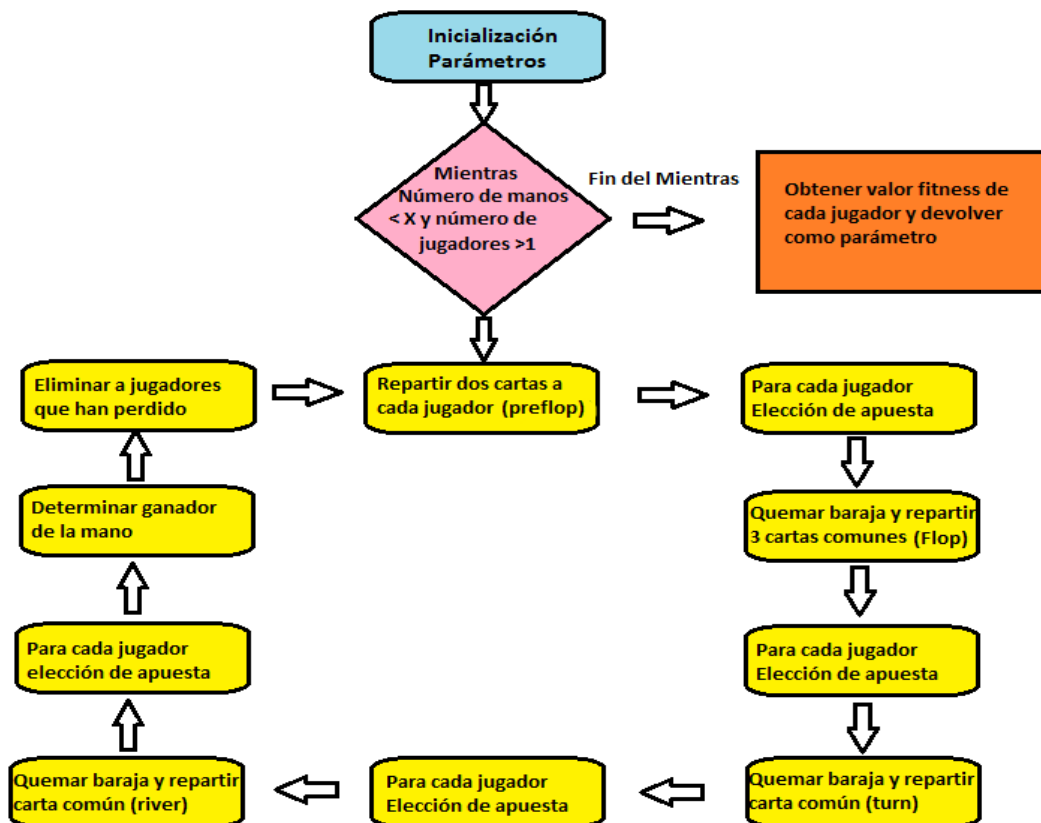
3.4 DISEÑO DE LA FUNCIÓN *FITNESS*

Como ya se ha comentado a lo largo de esta memoria, la función de evaluación o función *fitness* es la encargada de valorar la calidad de los jugadores según su estrategia, simulando una partida de Póker. De esta manera el Algoritmo Genético será capaz de seleccionar a los mejores individuos de cada generación.

El diseño de esta función es la parte más costosa, duradera y difícil de todo el trabajo, ya que es el pilar fundamental responsable del correcto funcionamiento del programa, así como el gran número de funciones auxiliares en las que se apoya. Cualquier fallo en la programación de la misma puede ser crítico.

La figura 11 es un diagrama de bloques que ayuda a entender el funcionamiento de la función *fitness*. A continuación se explica paso a paso cada uno de sus bloques.

Figura 11





Inicialización de Parámetros

En primer lugar se inicializan los parámetros que intervienen en la función. Estos son:

- Matrices auxiliares que guardan estrategia, bankroll y cartas de todos los jugadores. El bankroll de los jugadores está inicializado a 500 (que representa 500 fichas o 500 dólares, euros, etc).
- Matrices auxiliares que guardan los jugadores eliminados, los jugadores que han pasado, que han echado all-in, apostado o subido, etc.
- Parámetros auxiliares que guardan otro tipo de datos como porcentaje de jugadores que han visto un all-in, o que han subido etc.
- Valores de la ciega pequeña y ciega grande. (Ciega pequeña vale 5, y ciega grande vale 10, por tanto la apuesta mínima será de 10).

Condición de parada

El diagrama de flujo, que se corresponde con una mano de Póker, se repite mientras se cumplan dos condiciones: O que quede un único jugador ganador o que transcurra un número máximo de manos que varía dependiendo de la simulación del algoritmo (normalmente suelen ser 50 manos). Este límite se pone debido a que cuando transcurren varias generaciones en el Algoritmo Genético, la población está tan igualada (población de superindividuos), que las partidas de Póker pueden llegar a tener un número demasiado elevado de manos o incluso existe la posibilidad de que no acaben.

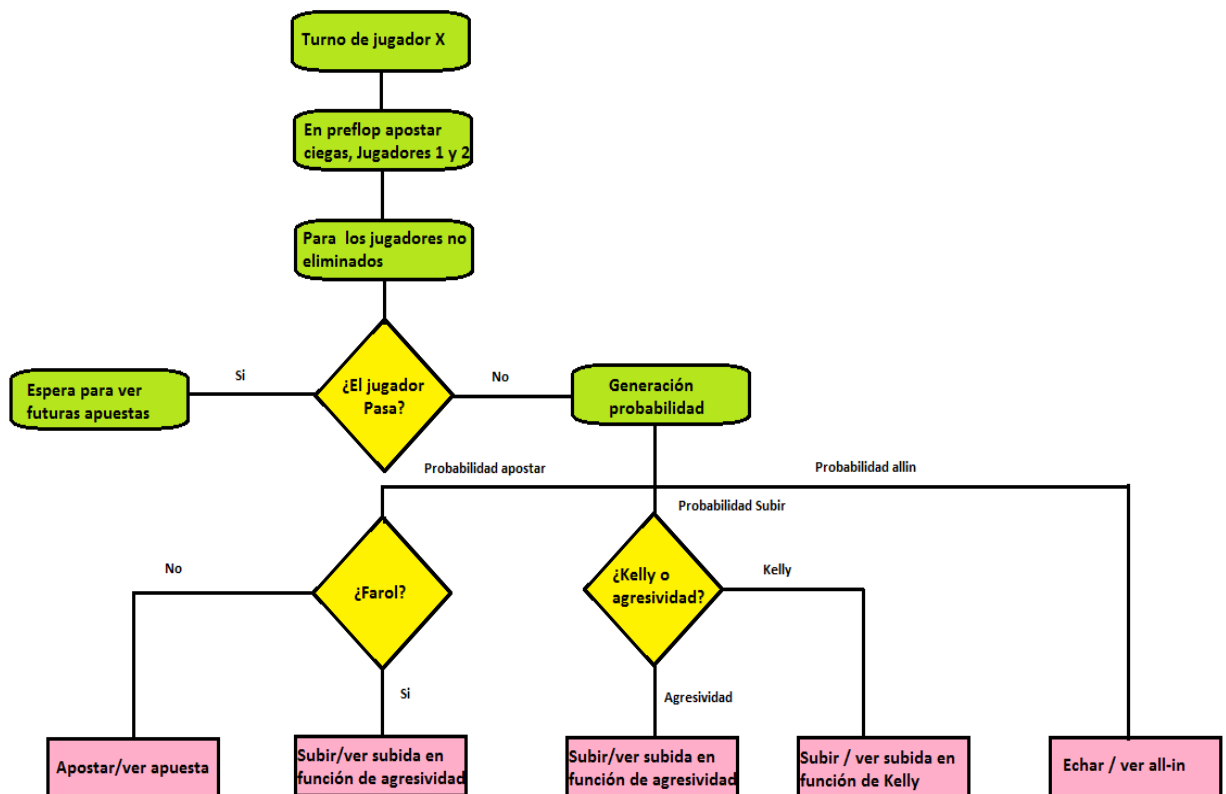
Rondas

Los bloque de: repartir dos cartas a cada jugador, quemar baraja (eliminar del juego la carta situada en la primera posición) y repartir tres cartas comunes, quemar baraja y repartir carta común y quemar baraja y repartir carta común se corresponden con las distintas rondas del juego: Preflop, Flop, Turn y River sucesivamente.

Elección de apuesta

Este bloque es común independientemente de la ronda que se esté jugando. Determina el tipo de acción que realizará cada jugador (pasar, apostar, ver apuesta, subir, ver subida, echar all-in o ver all-in).

Figura 12



La realización de las decisiones se determina para cada caso generando un número aleatorio del 0 al 100. A continuación se compara dicha cifra con el gen o los genes correspondientes de la estrategia y se realiza la decisión adecuada en cada situación.

La probabilidad de conseguir cierta jugada se estima como se explicó en el apartado *Probabilidades en el Póker*.



Determinación del ganador de la mano

El siguiente bloque, una vez realizadas todas las apuestas en las cuatro rondas de las que consta una mano, consiste en determinar el ganador.

Este paso es muy sencillo gracias a la organización matricial de las jugadas que se explicó en el apartado *Identificación de la Jugada*.

Consiste en identificar la combinación de cartas mayor conseguida para cada jugador y otorgarle la siguiente puntuación:

Puntuación =

$$= 10 * \text{Valor de la Jugada} + 0.1 * \text{Carta alta} + 0.001 * 2^{\circ} \text{ Carta alta}$$

Donde el valor de la jugada es del 1 al 10 en orden ascendente (1 carta alta, 2 pares...).

Recurriendo a los ejemplos que se propusieron anteriormente, los siguientes jugadores tendrían una puntuación de:

Figura 9

1º Parámetro	2º Parámetro	3º Parámetro	4º Parámetro	
Cartas Jugada	Cartas Mano	Carta Alta	Carta Alta 2	
0	0	0	0	0 Escalera Real
3	1	9	7	7 Escalera Color
0	0	0	0	0 Póker
0	0	0	0	0 Full
4	2	9	7	7 Color
5	2	10	9	9 Escalera
0	0	0	0	0 Trío
0	0	0	0	0 Dobles Parejas
0	0	0	0	0 Pareja
1	1	10	9	9 Carta Alta

$$\text{Puntuación} = 10 * 5 (\text{Escalera}) + 0.1 * 10 + 0.001 * 9 = 51.009$$



Figura 10

1º Parámetro	2º Parámetro	3º Parámetro	4º Parámetro	
Cartas Jugada	Cartas Mano	Carta Alta	Carta Alta 2	
0	0	0	0	0 Escalera Real
0	0	0	0	0 Escalera Color
0	0	0	0	0 Póker
5	1	11	6	6 Full
0	0	0	0	0 Color
2	1	13	11	11 Escalera
3	1	6	0	0 Trío
4	1	11	6	6 Dobles Parejas
2	0	11	0	0 Pareja
1	1	13	0	0 Carta Alta

$$Puntuación = 10 * 7 (Full) + 0.1 * 11 + 0.001 * 6 = 71.106$$

Ganará siempre el jugador que más puntuación tenga. Con este sistema se detectan fácilmente los empates (al coincidir puntuaciones). En dicho caso se recurre al valor de la siguiente jugada más alta, hasta alcanzar el desempate.

El valor del As está codificado como el número 14 (únicamente en esta matriz), ya que al ser la carta más alta, tiene que ser superior en valor a las demás.

Obtención del valor fitness de cada jugador

Una vez concluida la partida de Póker, ya sea porque queda un único ganador o porque ha transcurrido el número de manos establecido, es necesario establecer un valor *fitness* para cada individuo, para que el Algoritmo Genético pueda seleccionar a los mejores para su posterior reproducción y mutación.

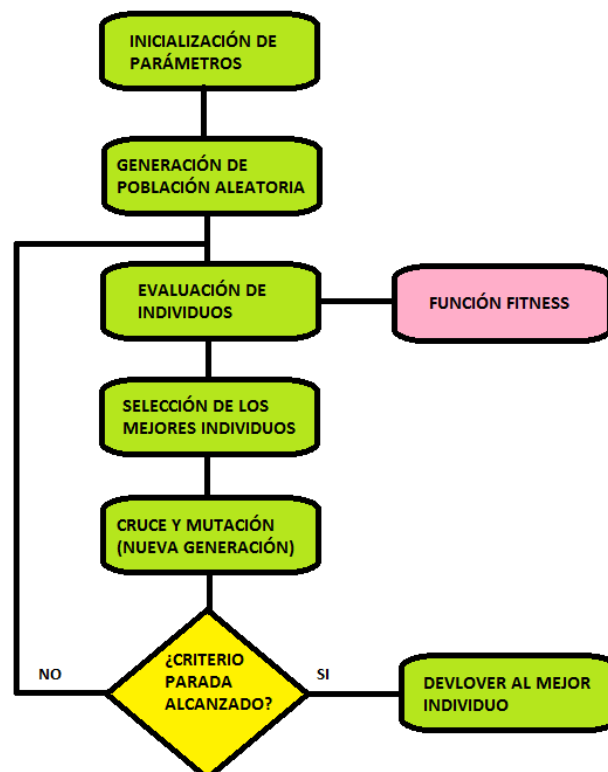
Este valor será la cantidad de fichas ganadas por cada jugador en la partida, ya que es muy intuitivo; los mejores jugadores son los que más dinero obtienen. Siendo nueve jugadores con 500 fichas cada uno, el máximo valor de *fitness* será de 4.500 en una partida en la que hay un solo ganador y todos los demás perdedores (con un valor de 0 cada uno, que es el mínimo).

3.5 DISEÑO DEL ALGORITMO GENÉTICO

Normalmente los Algoritmos Genéticos suelen tener un esqueleto común para la resolución de los problemas a los que están aplicados, variando en cada caso los parámetros más decisivos para su correcto funcionamiento. Estos son: longitud de la cadena genética, probabilidad de cruce, probabilidad de mutación, condición de parada (número de generaciones o porcentaje mínimo de convergencia) y número de población de cada generación.

A continuación, como muestra la figura 13 se propone un diagrama de bloques del algoritmo con la correspondiente explicación de cada bloque.

Figura 13





Inicialización de parámetros

Los valores de los parámetros mencionados en la página anterior son los siguientes:

- Longitud de cadena: 160, que coincide con la longitud de la estrategia que juega el papel de cadena genética.
- Probabilidad de cruce: 100%. Con este parámetro se reemplaza a toda la generación por sus descendientes sin dejar ningún superviviente de una generación a otra. Así nos aseguramos que no se reproduzcan padres con hijos (hecho que debe evitar en un Algoritmo Genético).
- Probabilidad de mutación: Varía según cada simulación. Normalmente con valores entre el 0.01% y 0.3%. Cuanto más alto sea dentro de este rango más oscilaciones se producen en el porcentaje de convergencia del algoritmo (del cual se hablará posteriormente), pero aumenta la diversidad genética a lo largo de las generaciones.
- Tamaño población: Si recordamos, un valor adecuado es entre una vez y dos veces la longitud de la cadena en binario. En nuestro caso, al ser números en base 100, se necesitarían poco menos de 7 bits para codificar cada gen. El Algoritmo está diseñado para que sea múltiplo de 900. Puede variar según la simulación (900, 1800 o 3600), que haciendo cuentas se adecúa con su similar en binario. Cuanto más alto sea el valor, más probabilidades hay de encontrar buenos resultados, pero el tiempo de ejecución aumenta.
- Condición de parada: El Algoritmo suele converger en casi todas las simulaciones rápidamente. A partir de la generación 40, más del 80% de la población obtiene los mismos resultados de evaluación. A partir de ese momento la convergencia crece muy lentamente. Para nuestro caso 100 generaciones es un número para alcanzar una convergencia estable y razonable (en torno al 90%, en muchos casos supera el 95%).

Generación de la población aleatoriamente

La población se genera aleatoriamente individuo por individuo, tal y como se indica en el apartado *Diseño de la estrategia, Inicialización de la estrategia*.



Evaluación de los individuos

Se juega por cada generación un número de partidas igual al tamaño de la población dividido entre 9 (ya que son grupos de 9 personas). A cada individuo se le evalúa con un número de 0 a 4500 equivalente al dinero que ha ganado en una partida de Póker. En las primeras generaciones se dan valores muy dispersos: muchos jugadores son perdedores (con un *fitness* de 0), en torno a un 30% de los jugadores tienen un *fitness* de 500 (que será el objetivo de convergencia, ni ganar ni perder, cuando todos los jugadores estén igualados), y muchos jugadores tienen evaluación superior a 500, con valores muy esparcidos.

Con el paso de las generaciones disminuye el número de ganadores y perdedores, y aumenta el número de jugadores que tienen un *fitness* en torno a 500.

Selección de los mejores individuos

El método de selección se conoce como método de los torneos, explicado en el capítulo *Algoritmos Genéticos*, que se basa en hacer competir a los distintos individuos y seleccionar a los que obtengan mejor resultado. Una de las preguntas que surgen a la hora de diseñar el algoritmo es: ¿Cuántos individuos debo seleccionar para su reproducción?

Está claro que no se debe seleccionar únicamente los ganadores de cada partida, ya que es un porcentaje de población muy pequeño que produciría una convergencia muy prematura con mucha pérdida de información y resultados falsos.

Ya que para ganar al Póker no se necesita sólo una buena estrategia sino también suerte, y no es deseable tener una pérdida de información elevada, el método de selección se basa en escoger a la mitad de la población que haya obtenido mejor *fitness* (que siempre será de 500 hacia arriba). Hay muchos de estos individuos que no se pueden denominar ganadores, pero sí no perdedores.

De esta manera, con el paso de las generaciones se obtendrán los mejores resultados posibles manteniendo gran diversidad genética.

Cruce y mutación

Una vez seleccionados los individuos mejores de cada generación deben ser cruzados y mutados para generar una nueva población que sustituirá a su predecesora.

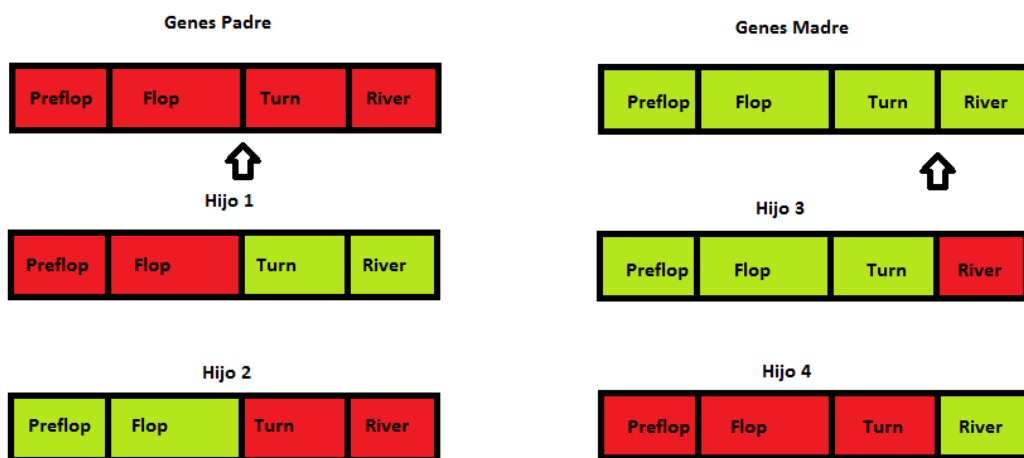
Al estar seleccionada la mitad de los individuos, cada pareja deberá tener cuatro hijos para duplicarse en número y volver al tamaño de población original.

Se establecen tres puntos de cruce (los genes que separan el Preflop del Flop, el Flop del Turn, y el Turn del River). Estos son el 26, 79 y 124 sucesivamente. De estos tres puntos distintos se seleccionan 2 distintos aleatoriamente.

Se realiza de esta forma para no mezclar información, perteneciente a una misma ronda, de distintos individuos. Si recordamos, la probabilidad de all-in más la de subir sumada con la de apostar (esta última no está implícita en la estrategia), debe ser igual al porcentaje de actividad, y éste último más el de pasar debe ser igual al 100%. Si no se separasen las cuatro rondas por estos tres puntos de cruce, ya no se cumplirían las sumas de los porcentajes anteriores.

La figura 14 ayuda a comprender mejor cómo se lleva el proceso de cruce.

Figura 14





La mutación está pensada para cambiar un bit de 0 a 1 o de 1 a 0 en binario, y esto repercute en mayor o menor medida según sea un bit más o menos significativo.

Para nuestro caso, si se cumple la probabilidad de mutación, se genera un número aleatorio del 1 al 160 que indica el gen que va a cambiar. Posteriormente ese gen mutará sumando o restando un 10% a la probabilidad que representa (decidido por otro número aleatorio).

Como se comentaba anteriormente es importante que la probabilidad de all-in, subir y apostar, sean iguales a la de actividad. La mutación no origina problemas en este sentido, ya que al no estar implícita la probabilidad de apostar en la estrategia, si por ejemplo la probabilidad de subir aumenta, la de apostar disminuirá y viceversa.

Criterio de parada

El criterio de parada se alcanza cuando un número elevado de la población ha convergido (superior al 85% - 90 %), con un valor de *fitness* de 500.

El Algoritmo Genético no finaliza cuando se alcanza este porcentaje de convergencia, ya que en todas las simulaciones alcanza estos valores en generaciones no muy avanzadas debido a máximos locales. Es necesario establecer un número elevado de generaciones (en torno a 100 o 150 para poblaciones grandes), para que la convergencia sea estable y no presente picos.

Cuando finaliza la simulación de la última generación, se devuelve como solución al mejor individuo, el que mayor *fitness* tenga.

En el siguiente capítulo se explicará con más detalle por qué el algoritmo converge, qué pruebas se han realizado para comprobarlo, los resultados obtenidos y conclusiones.



4. RESULTADOS Y CONCLUSIONES

4.1 CONVERGENCIA

Uno de los hechos que demuestran que el Algoritmo Genético funciona correctamente es su convergencia. Esto se consigue con el paso de varias generaciones, cuando todos los individuos de una población alcanzan valores de *fitness* similares debido a que tienen una cadena genética muy parecida. Este es el resultado que deseamos obtener.

En el caso de diseñar estrategias óptimas en el Póker, es imposible converger a una única solución, es decir, que todos los individuos tengan exactamente la misma cadena genética, dado a que ésta es muy compleja. Además, la solución obtenida en cada simulación dependerá de la población inicial (siempre inicializada aleatoriamente), de los cruces y mutaciones, y de la buena o mala suerte que tengan los jugadores en las partidas.

Se realizan varias pruebas explicadas a continuación para comprobar la convergencia.

Histograma

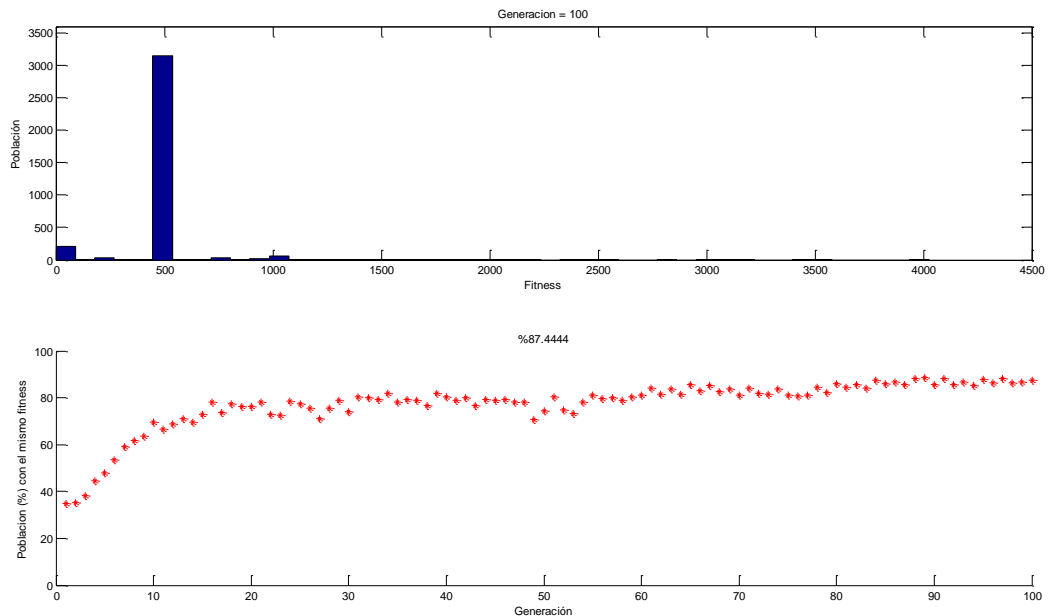
Un histograma es un diagrama de barras que muestra la evolución estadística de una variable, un conjunto de datos, etc.

En este caso se representa el número de la población (en el eje vertical) que alcanza los distintos valores de *fitness* (eje horizontal), separados en intervalos de 100 en 100.

Esta representación, que se actualiza cada vez que entra una nueva generación, se puede ver en vivo cuando el programa se está ejecutando.

La figura 15 es una captura del histograma en la última generación.

Figura 15



En la parte superior se representa el histograma. Como se puede observar en este caso, en una población de 3600 personas, más de 3000 han alcanzado la convergencia, con un valor de *fitness* de 500 (ni ganan ni pierden dinero). Este valor supone una convergencia del 87.44%.

El resto de jugadores tienen valores de *fitness* distintos a este número, que varían de 0 a 4500.

En la parte inferior de la figura se muestra la evolución de la convergencia con el paso de las generaciones. Para ello se determina cuál es el valor de *fitness* que ha alcanzado la mayor parte de los individuos, y qué porcentaje de la población representa.

Este diagrama evoluciona de forma similar en todas las simulaciones, lo que implica que el algoritmo funciona. La convergencia comienza con valores entre el 35% y 40% y comienza a subir bruscamente en la tercera o cuarta generación hasta la generación 20, a partir de la cual crece más lentamente hasta que se estabiliza.

Como se puede observar, el diagrama inferior presenta oscilaciones debidas a que en algunas generaciones hay más individuos ganadores y perdedores y se reduce ligeramente el número de los que alcanzan 500 de *fitness*.



Los resultados obtenidos con este experimento son razonablemente buenos. Una convergencia del 100% supondría una población sin ganadores ni perdedores; en tal caso no se podría devolver como resultado a un individuo ganador.

Las cifras de convergencia obtenidas en distintas simulaciones varían entre valores superiores al 85% e inferiores al 97%.

“Fotografía” de las estrategias de la última generación

Otra forma de comprobar el correcto funcionamiento del algoritmo, es tomar una “fotografía” de las estrategias de la última generación, cuando el algoritmo ha convergido. Esto se realiza con la función *“imshow()”* proporcionada por MATLAB, que muestra por pantalla imágenes en formato digital.

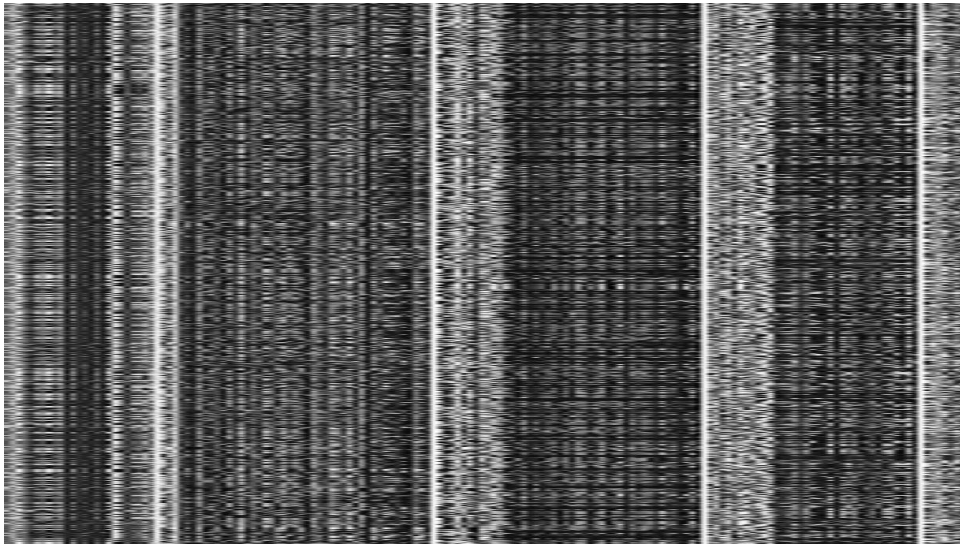
La fotografía es en escala de grises. Los valores más cercanos al negro representan probabilidades bajas, y los más cercanos al blanco representan probabilidades altas.

Es una imagen de tamaño: número de población píxeles verticales x 1600 píxeles horizontales.

Los píxeles horizontales representan los genes de la población (está redimensionada horizontalmente para que se pueda observar mejor), y los verticales representan a los distintos jugadores.

La figura 16 es ejemplo extraído de una simulación.

Figura 16



Las líneas verticales que se observan, corresponden a un mismo gen, para los distintos jugadores. Al no variar de color en el sentido vertical, indican un comportamiento común para toda la población, lo que demuestra la convergencia del algoritmo.

Varianza

La varianza es un parámetro estadístico que mide la dispersión de una variable aleatoria. En nuestro caso mide cuánto se alejan los jugadores en cada gen de la media de dicho gen que pertenece a la población.

Su fórmula matemática es:

$$\sum_{i=1}^{población} f(x_i) * (X_i - Ux)^2$$

Donde:

- $f(x)$ es la frecuencia de cada individuo (al ser N individuos, la frecuencia de cada uno será $1/N$).
- X es el gen al que le estamos calculando la varianza.
- Ux es la media de dicho gen.



Se calcula la varianza para todos los genes de la población, para observar lo alejados que están de la media de jugadores. Tiene valores que varían dependiendo de los genes, entre valores próximos a 10 en varianzas bajas y próximos a 800 en varianzas altas. Hay que tener en cuenta que la raíz de la varianza o desviación típica proporciona valores más aproximados sobre cuánto varía un porcentaje hacia arriba o hacia debajo de la media.

No se puede considerar ningún valor de la varianza demasiado elevado para decir que el algoritmo no converge.

Los valores de las varianzas de las simulaciones realizadas se encuentran en el adjunto 2 Excel *simulaciones.xml* que se proporciona.

Media

El último experimento que se realiza es la media de los genes de toda la población. Los valores obtenidos, para cada caso están en el adjunto 2 *simulaciones.xml*.

Aunque la varianza ya nos da información de lo cercanos que están unos individuos de otros, la media nos proporciona datos sobre una estrategia global que sigue esta población.

La comparación gen a gen entre la media y la estrategia del individuo ganador no dista demasiado. Para obtener valores más globales, se calcula la distancia media de todos los genes para el ganador y para la población entera. Sus valores se encuentran, dependiendo del caso entre el 15% y el 19%.

Por tanto significa que en media, cada gen de un individuo dista de la media en un 15% o 19%. En algunos genes estas distancias son mucho menores (en torno al 5%) y en otros más elevada (en torno al 25%).



4.2 RESULTADOS

Los resultados de las estrategias ganadoras se proporcionan en el adjunto 2 *simulaciones.xml* debido a que se visualizan mejor en una hoja Excel.

En este apartado se explica con detalle la valoración y comparación de cada uno de los individuos vencedores.

Se proporciona el resultado correspondiente a tres simulaciones, a partir de ahora *Estrategia A*, *Estrategia B* y *Estrategia C*.

La *Estrategia A* se corresponde con una simulación de 900 de población, 50 manos por partida como máximo, una probabilidad de cruce del 100%, probabilidad de mutación del 0.2% y 100 generaciones.

La *Estrategia B* se corresponde con una simulación de 1.800 de población, 50 manos por partida como máximo, probabilidad de cruce del 100%, probabilidad de mutación del 0.3% y 100 generaciones.

La *Estrategia C* se corresponde con una simulación de 3.600 de población 50 manos por partida como máximo, probabilidad de cruce 100%, probabilidad de mutación 0.1% y 150 generaciones.

La elección de estos valores no tiene ningún por qué, excepto en la *Estrategia C* que hay más generaciones porque la población es mayor, el algoritmo tarda más en converger. El resto de parámetros solo afectan a la convergencia que influye en el resultado final, y estas pequeñas variaciones no perturban a la convergencia de una simulación a otra.

De esta se demuestra la robustez de la solución frente a los parámetros inicializados en el Algoritmo Genético, ya que siempre se llega a parecida solución.

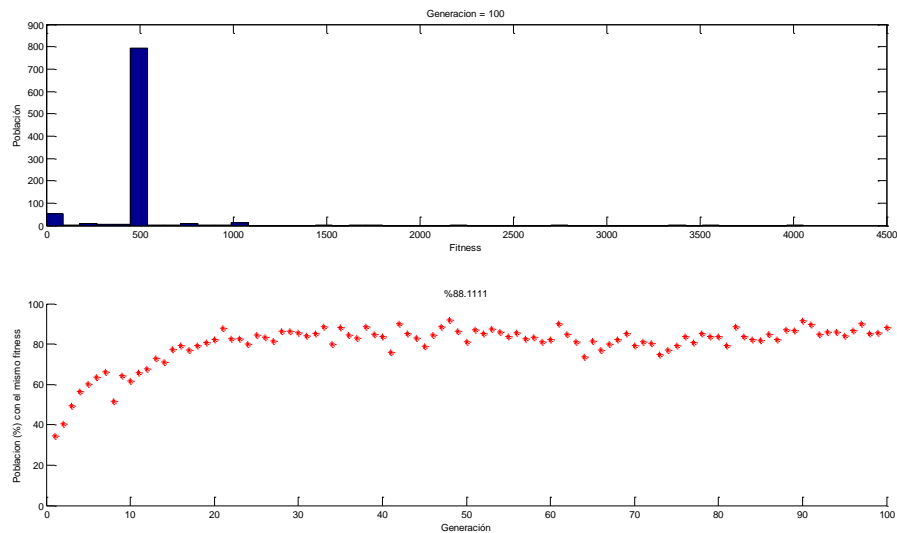
A continuación se muestran para cada una de las tres simulaciones los resultados de los experimentos aplicados para comprobar la convergencia, explicados en el apartado anterior.

Resultados, convergencia

Estrategia A: Convergencia conseguida 88,11%

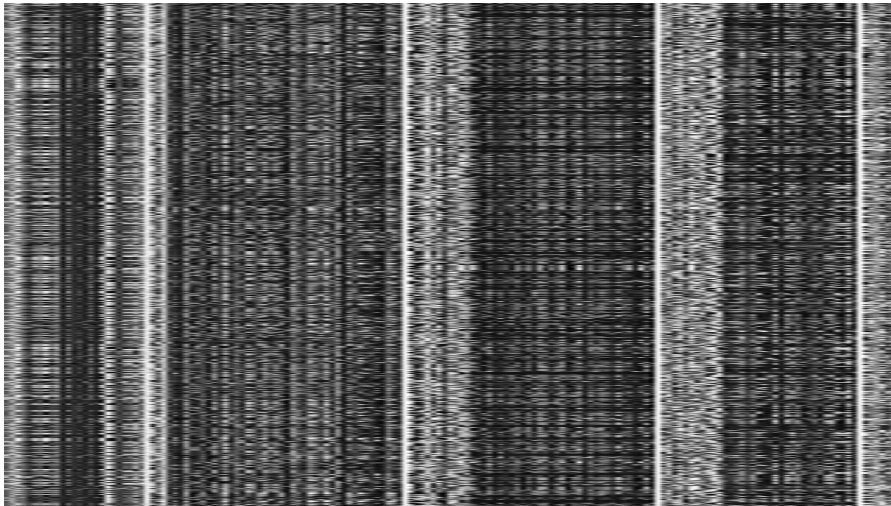
Histograma

Figura 17



“Fotografía” de la población

Figura 18



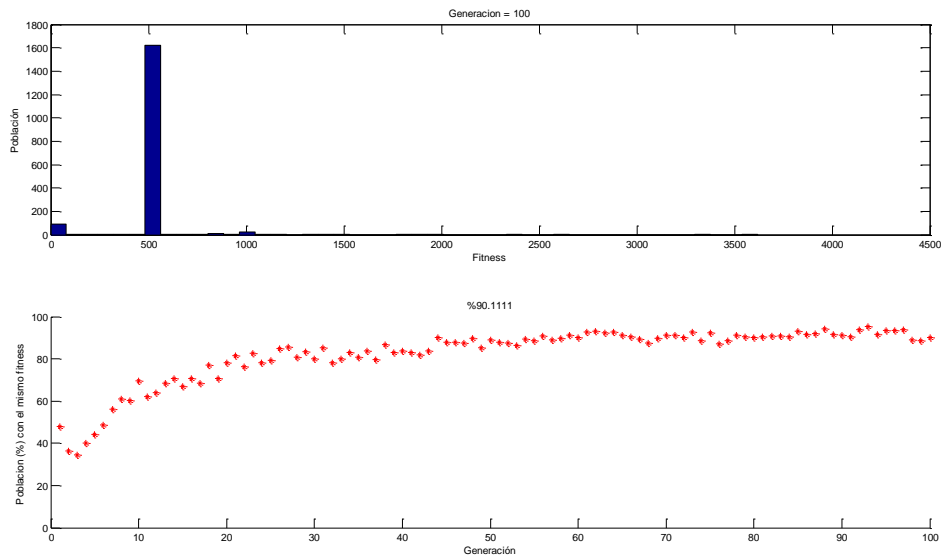
Varianza: en el adjunto 2.

Media: Esta simulación ha conseguido una distancia de 15,45 de media para el ganador, y una distancia de 18,7 de media para la población.

Estrategia B: Convergencia conseguida 90,11%

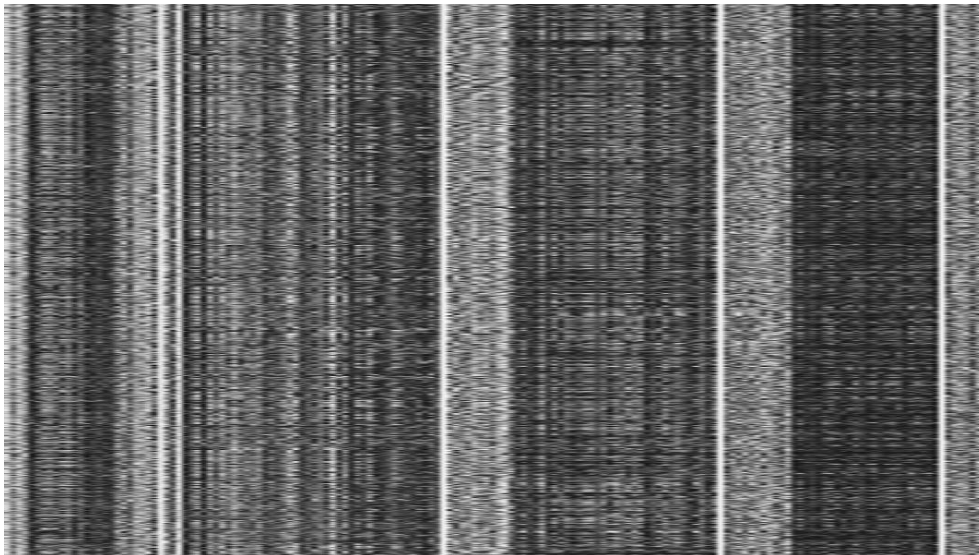
Histograma

Figura 19



“Fotografía” de la población

Figura 20



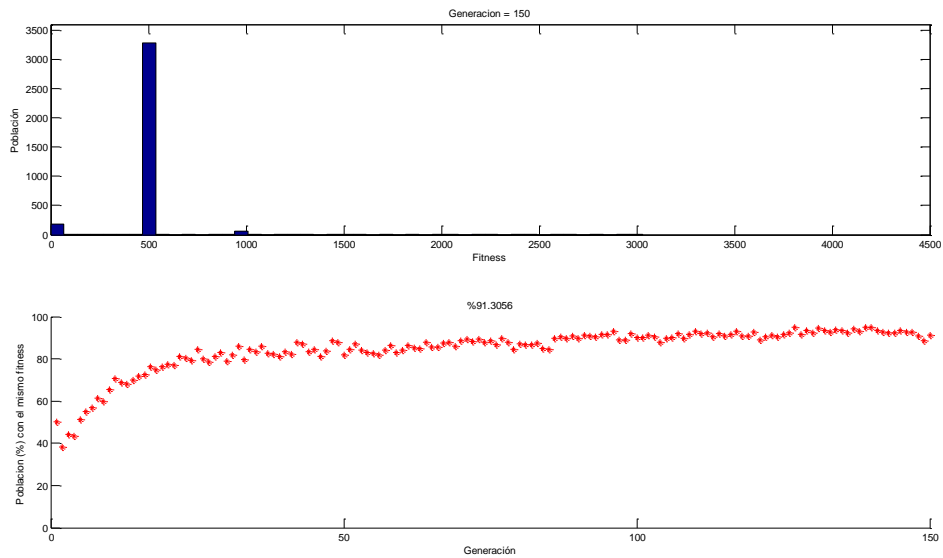
Varianza: en el adjunto 2.

Media: Esta simulación ha conseguido una distancia de 16,9 de media para el ganador, y una distancia de 18,79 de media para la población.

Estrategia C: Convergencia conseguida 91,3%

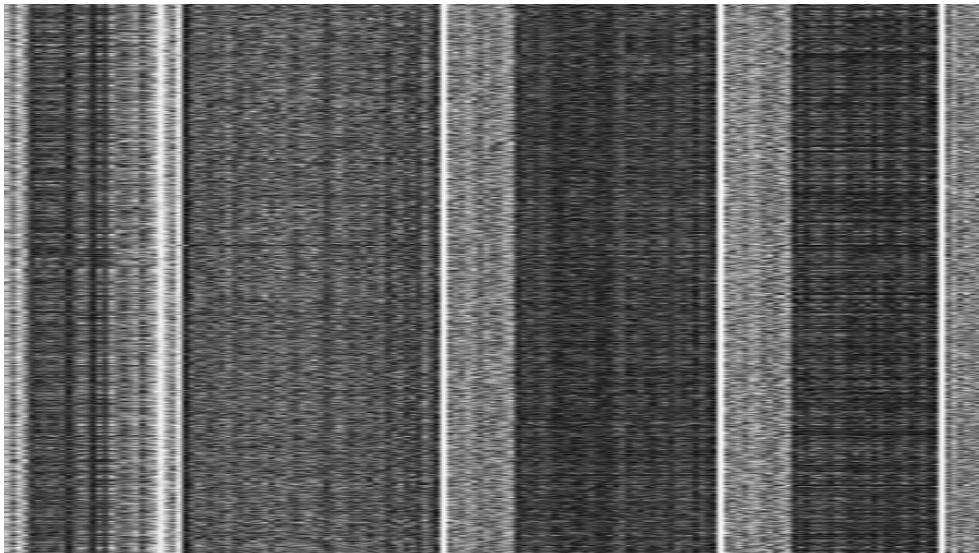
Histograma

Figura 21



“Fotografía” de la población

Figura 22



Varianza: en el adjunto 2.

Media: Esta simulación ha conseguido una distancia de 19.59 de media para el ganador, y una distancia de 19.5 de media para la población.

Nótese que cuanto mayor es el número de población, los valores de convergencia aumentan, debido a que el algoritmo está funcionando mejor. Esto no ocurre siempre, pero generalmente si se cumple.

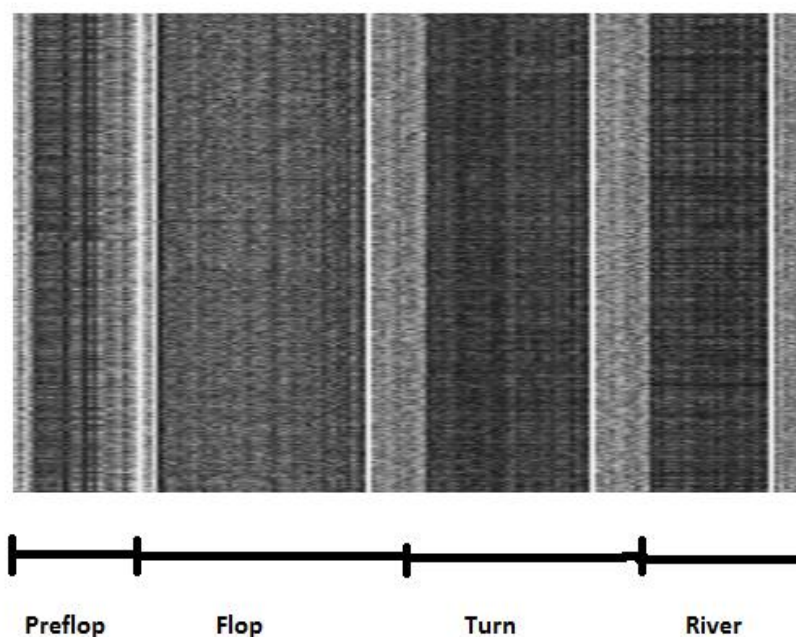
Es importante tener en cuenta las siguientes consideraciones. El hecho de que la convergencia aumente no significa que la media de la distancia a la media de los genes disminuya. Este último dato es orientativo, ya que está teniendo en cuenta a toda la población, en la que puede haber individuos muy buenos y muy malos.

Cuando la convergencia es alta no significa que el resultado obtenido sea mejor que el de otra simulación cuya convergencia sea menor. Como ya se ha explicado, el resultado es el mejor jugador de la última generación. Aunque su estrategia sea buena, no implica que no hayan pasado mejores jugadores por el algoritmo, simplemente puede haber tenido más suerte con las cartas, o haber jugado contra peores jugadores.

Resultados, comparaciones

A continuación procedemos a comparar las figuras 18, 20 y 22, de las que se pueden sacar conclusiones sobre cómo juega la población al Póker. Para ello la figura 23 muestra cómo se dividen los genes.

Figura 23





Se llega a la conclusión de que las divisiones son como se muestra en la figura 23, observando los genes de las estrategias de los ganadores (adjunto 2, simulaciones.xml) y por la longitud genética correspondiente a cada ronda.

Para todas las simulaciones de las estrategias (no solo para las tres simulaciones que se proporcionan, si no para todas las que se han simulado), cada ronda (Preflop, Flop, Turn y River) comienza con una franja gris clara (que indica probabilidades altas de farol, actividad, agresividad y de apostar siguiendo el criterio de *Kelly*) y terminan con una línea gris, en algunos casos más clara y en otros más oscura, que indica las probabilidades de ver subida y ver all-in dependiendo del número de jugadores que lo hayan visto, y del dinero que tengan.

El Preflop termina con una línea blanca que indica la probabilidad de retirarse si el jugador tiene poco dinero (en ningún resultado de ninguna simulación ha bajado del 85%, siendo valores típicos del 96% al 100%). Este dato es el más relevante para probar que en distintas simulaciones, aunque los resultados no son iguales sí son parecidos, ya que es el que menos varía.

El resto de líneas blancas son la probabilidad de ver/echar all-in con escalera real que está inicializada al 100%, que en algún caso puede haber sido modificado por mutación, pero por lo general estos valores son altos.

Comparando las tres simulaciones proporcionadas, en las figuras 18, 20 y 22, se puede observar que las probabilidades de subir/ ver subida y echar/ver all-in (franja gris más oscura que ocupa casi todo el espacio de cada ronda) no son muy altas (recordad que la probabilidad de apostar/ver apuesta, no está implícito en la estrategia, y las tres probabilidades junto con la de pasividad suman 100%, el color blanco). Estas probabilidades disminuyen a medida que aumenta la ronda.

También se observa cómo los cuatro primeros genes (actividad, agresividad, farol y *Kelly*) son más elevados en el Preflop y Flop que en el Turn y River.

La *Estrategia A* tiene jugadores muy activos a la hora de subir y echar all-in y otros jugadores que no lo son (se observan en las líneas o franjas negras y blancas horizontales), mientras que las Estrategias B y C son más homogéneas.

Todas estas fotografías son similares, tienen la misma estructura en todos los casos.

A continuación, en las figuras 24,25, 26 y 27 aparecen datos relevantes de las tres simulaciones.

Figura 24

	Preflop	Flop	Turn	River
Farol				
Estrategia A	39	72	17	50
Estrategia B	84	69	49	93
Estrategia C	78	92	89	2
Agresividad				
Estrategia A	45	9	31	9
Estrategia B	73	26	79	45
Estrategia C	2	60	46	77
Actividad				
Estrategia A	100	66	72	87
Estrategia B	69	94	98	64
Estrategia C	45	100	56	95
Kelly				
Estrategia A	55	34	69	91
Estrategia B	27	6	21	55
Estrategia C	98	0	4	23
P. Retirarse				
Estrategia A	88			
Estrategia B	96			
Estrategia C	99			

Como se ha comentado anteriormente, el dato más llamativo común a todas las simulaciones es la probabilidad de retirarse, también visible en las fotografías de las estrategias, con valores muy elevados.

La probabilidad de farol, de agresividad y de actividad es bastante elevada en los tres individuos. Esto indica que el perfil de todos los individuos es generalmente agresivo.

El *individuo A* es ligeramente menos agresivo y farolero que los otros dos, y aplica el criterio de *Kelly* con más frecuencia. Se puede considerar el criterio de *Kelly* contrario a la agresividad. Aunque este criterio de como resultado valores elevados en ocasiones, un individuo que suele ser agresivo no se preocupa por métodos más científicos, como es el de *Kelly*.



Figura 25

	Preflop	Flop	Turn	River
Ver Subida 75%-100%				
Estrategia A	12	1	73	34
Estrategia B	73	90	8	37
Estrategia C	52	8	30	56
Ver Subida 50-75%				
Estrategia A	51	11	40	91
Estrategia B	2	9	75	79
Estrategia C	13	12	66	68
Ver Subida 25-50%				
Estrategia A	31	65	37	66
Estrategia B	28	51	17	83
Estrategia C	88	30	33	28
Ver Subida 0-25%				
Estrategia A	6	99	53	27
Estrategia B	82	59	16	99
Estrategia C	89	28	87	42
Ver Allin + Bankroll				
Estrategia A	17	37	79	39
Estrategia B	66	69	31	77
Estrategia C	89	94	10	56
Ver Allin - Bankroll				
Estrategia A	14	97	41	16
Estrategia B	13	36	13	49
Estrategia C	95	25	22	83
Ver allin + jugadores ven				
Estrategia A	63	14	95	75
Estrategia B	1	43	86	21
Estrategia C	58	88	34	7



En general, los datos que nos muestra la figura 25, son cifras bastante elevadas. Son buenos resultados para mantener a una población bastante activa y agresiva (como la que tenemos en los resultados). La probabilidad de ver una subida o un all-in depende de las cifras anteriores y de la jugada que tenga el individuo.

Por lo general, las probabilidades son mayores en el River que en las otras tres rondas. Es un resultado lógico, ya que un jugador que se ha mantenido activo durante toda una mano, debería mantenerse activo también en la última ronda, para no perder las cantidades apostadas anteriormente.

Figura 26

	P. Subir			
	Preflop	Flop	Turn	River
Carta Alta				
Estrategia A	36	19	55	56
Estrategia B	9	39	19	42
Estrategia C	1	34	4	0
Pares				
Estrategia A	77	16	18	74
Estrategia B	50	84	6	26
Estrategia C	23	25	53	63
Pares Altos				
Estrategia A	66	12	0	55
Estrategia B	21	3	33	16
Estrategia C	36	14	54	8
D. Parejas				
Estrategia A		55	37	31
Estrategia B		79	15	40
Estrategia C		70	3	30
D. Parejas A				
Estrategia A		42	10	11
Estrategia B		27	65	27
Estrategia C		52	22	67

Figura 27

	P. Allin	Flop	Turn	River
Carta Alta				
Estrategia A	54	25	0	10
Estrategia B	15	28	35	4
Estrategia C	25	40	45	82
Pares				
Estrategia A	11	48	38	7
Estrategia B	10	7	89	38
Estrategia C	22	37	1	4
Pares Altos				
Estrategia A	29	27	40	8
Estrategia B	28	87	45	38
Estrategia C	2	21	1	75
D. Parejas				
Estrategia A		4	18	29
Estrategia B		10	76	19
Estrategia C		15	41	17
D. Parejas A				
Estrategia A		16	26	66
Estrategia B		62	2	13
Estrategia C		47	18	9

Las probabilidades de subir/ver subida y echar/ ver all-in que se muestran en las figuras 26 y 27 son un poco dispersas, pero hay que tener en cuenta que estas probabilidades sumadas junto con las de apostar y pasividad tienen que sumar el 100%, por lo tanto no deberían ser demasiado altas.

Sin embargo los resultados demuestran que en muchos casos estas cifras son elevadas, debido a que la población tiene un perfil activo y agresivo como ya se ha comentado anteriormente.



No hay que fiarse mucho de estas cifras, solo son orientativas, por la razón de que en una partida de póker la jugada más común y con la que más se suele ganar es con pares altos, dobles parejas o tríos como mucho. Las demás combinaciones de cartas se consiguen con una probabilidad muy baja.

Es decir, si por ejemplo en la estrategia de un jugador nos encontramos una probabilidad baja de subir o echar all-in con escalera de color, a primera vista puede parecer contradictorio, porque es una jugada que gana a todas las demás excepto a escalera real, pero hay que tener en cuenta que esa jugada le habrá salido a dicho individuo muy pocas veces en las partidas simuladas, o incluso ninguna vez.

4.3 CONCLUSIONES

Como ya sabemos, los Algoritmos Genéticos son la mejor solución para resolver problemas cuya naturaleza es muy compleja, y no existe otro método específico para resolverlos.

Como se explicó en el capítulo *Algoritmos Genéticos*, una mala decisión a la hora de dar valores a las probabilidades de mutación y reproducción, así como el número de generaciones, puede ser fatal para la convergencia del algoritmo. En nuestro caso estos valores, dentro de un rango lógico, no influyen.

Una vez que el Algoritmo ha convergido correctamente, encontramos la estrategia ganadora, o al menos la no perdedora. Es un error pensar que utilizando este método vamos a encontrar siempre la mejor solución, o que en varias simulaciones encontremos soluciones idénticas gen a gen. Esto no ocurre a menos que sepamos a priori cual es la estrategia óptima.

Aparte, en un juego como es el Póker, no todo es tener una buena estrategia, sino la suerte es un tema fundamental para ganar, y eso no se puede controlar.

Por tanto, la solución hallada dependerá tanto de la población inicial como de las probabilidades de mutación, el número de generaciones, el correcto diseño de la función fitness y de los genes, etc.

Los Algoritmos proporcionan pues, la mejor o una de las mejores soluciones para una determinada población, que vendrá influida por los factores comentados anteriormente.



No hay que pensar entonces que se puede ir a un casino con una de estas estrategias teniendo asegurado el éxito, porque aparte de lo mencionado en la página anterior toda estrategia tiene una contraestrategia, mejor o peor en calidad, que la puede ganar. No existe una estrategia infalible, imposible de vencer por otra.

El hecho de encontrar estrategias óptimas en el Póker es algo muy complicado y perseguido, ya que es un juego que mueve cantidades elevadas de dinero. Para ello se utilizan los Algoritmos Genéticos, pero para situaciones más concretas, con una determinada jugada o ante una determinada situación, y no para partidas enteras, ya que de esta manera se tiene mejor controlado al algoritmo y las soluciones encontradas son más precisas.

Líneas futuras

Una de las posibles mejoras que se podrían aplicar a este software, sería ampliar la función que calcula las probabilidades de ganar con ciertas jugadas en el Póker aplicadas al criterio de Kelly. Como se explicó en su correspondiente apartado, se utiliza una versión simplificada del cálculo de las probabilidades, ya que es un tema muy complejo, ya que el hecho de ganar depende tanto de tus cartas, como las del resto de jugadores y las que se sitúan en la mesa.

Factores como la suerte o la calidad con la que un individuo farolea o detecta faroles no figuran en la cadena genética. Otra posible mejora sería incluirlos aunque son difíciles de implementar, ya que el factor suerte o calidad con la que se farolea son variables más abstractas.



Adjuntos

Adjunto 1: Software. Para ejecutar, abrir con MATLAB la función **algoritmoGenético**.

Adjunto 2: Resultados obtenidos en el archivo ***simulaciones.xmlx***

Adjunto 3: Resumen del trabajo.



Referencias

- [1] *Reglamento El Póker Texas Hold'Em*, Fournier [www.nhfournier.es, 05/08/2012].
- [2] *Póker Texas Hold'Em* [wikipedia.es, 05/08/2012]
- [3] *Algoritmos Genéticos*, Javier Ventoso Reigosa (Artículo publicado en la revista Sólo Programadores Num. 89).
- [4] *Algoritmos Genéticos*, Pedro Isasi (Asignatura Computación Biológica. Departamento de Informática Universidad Carlos III de Madrid).
- [5] *Algoritmos Genéticos* [wikipedia.es 06/08/2012]
- [6] *La teoría de los juegos, sus aplicaciones en economía y metodología*, José Luis Ferreira García (Departamento de Economía Universidad Carlos III de Madrid)
- [7] *Comprendiendo la teoría de juegos y el Hold'Em*, Bryce Paradis ; Douglas Zare.
- [8] *Teoría de los Juegos* [wikipedia.es 06/08/2012]
- [9] *Póker y negocios* [www.euribor.com.es 08/01/2009]
- [10] *Póker online* [wikipedia.es 09/08/2012]
- [11] *Guía de Apuestas Deportivas*. Juan Carreño Garro
- [12] www.poquer.com.es/jugadores-poker