

Memoria de investigación

Computación Evolutiva

Jorge Vázquez Acevedo
Jaime Miguel Hernández

Contenido

Tema a tratar y Objetivos	2
Técnica para abordar el problema / Conceptos previos: algoritmos genéticos	3
Genoma	3
Función de Evaluación	4
Operadores de selección	4
Operadores de cruce	5
Operadores de mutación	5
Adaptación del problema al algoritmo genético	6
Criatura	6
Genoma	6
Combate	6
Función de evaluación	7
Planteamiento Inicial	8
Problemas y soluciones	9
Resultados finales	11
Conclusiones y Trabajo Futuro	17
Objetivos alcanzados	17
Objetivos no alcanzados	17
Posibles mejoras	17
Posibles aplicaciones	17
Conclusión final	18

Tema a tratar y Objetivos

El trabajo realizado se basa en una investigación sobre un modelo evolutivo en el que simulamos un ecosistema de criaturas, las cuales combaten para determinar cuál es más la fuerte. En esta investigación propondremos un modelo del ecosistema, con una herramienta para simular los combates, y utilizando un algoritmo evolutivo trataremos de encontrar criaturas cada vez mejores (de acorde al mencionado sistema de combate).

Durante el desarrollo de la investigación introduciremos cambios en el algoritmo evolutivo, que permitirá encontrar resultados diferentes, con una convergencia más rápida o con mayor diversidad, entre otros aspectos importantes. Así mismo, analizaremos que resultado tienen estos cambios, si el resultado es el que esperábamos cuando se pensó el cambio, y si mejora alguno de los aspectos mencionados: diversidad, velocidad de convergencia, exploración de otros mínimos relativos, “clusterización” de los individuos en el dominio de búsqueda.

Técnica para abordar el problema / Conceptos previos: algoritmos genéticos

Antes de entrar en la investigación realizada, vamos a exponer brevemente algunos de los conceptos relacionados con los algoritmos genéticos y sus símiles con los procesos de evolución de la naturaleza.

En primer lugar, como su propio nombre indica, un algoritmo evolutivo es un algoritmo de búsqueda, basado en el proceso natural de evolución de criaturas, y en el concepto de supervivencia del más fuerte. Permite encontrar soluciones, de manera más óptima que si se hiciera por fuerza brutal, mediante la adaptación del problema a un genoma (una manera de describir las soluciones), una función de evaluación (que permite, valga la redundancia, evaluar cuánto de buena es una solución); unos operadores de selección (cómo escogemos, dependiendo del valor obtenido en la función de evaluación, qué criaturas van a perdurar en la siguiente generación); de cruce (la forma que tienen las soluciones de entremezclarse en su afán de mejorar), y de mutación (permitiendo a las soluciones salir de máximos relativos en su búsqueda de absolutos).

Quizá la diferencia más significativa en el uso que hacemos nosotros del algoritmo, con respecto a lo que ocurre en la naturaleza, es que trabajamos con generaciones en tiempo discreto. Es decir, las diferentes criaturas no van naciendo y muriendo de manera simultánea, sino que todas van pasando por las mismas fases al mismo tiempo. Por lo demás, conceptualmente hay muchísimas similitudes entre la evolución natural y los algoritmos evolutivos utilizados.

GENOMA

Conjunto de datos que representa al individuo: que características va a tener, como se va a comportar... No hay una manera fija de expresar esta información, pudiendo ser, por ejemplo, una lista de valores enteros, valores en coma flotante, caracteres, información binaria, o incluso mezcla de todos ellos.

El proceso de definición del genoma representa un paso muy importante en el desarrollo del problema, ya que dependiendo de cómo se representen los individuos será más fácil posteriormente controlar las dinámicas de convergencia o *clusterización* de los individuos. Si el genoma no está correctamente definido pueden darse casos, por ejemplo, de sobreajuste (los individuos aprenden a superar esa función de evaluación concreta, pero si la cambias por otra similar no se comportan de manera adecuada), o de no llegar a encontrar una solución subóptima (una solución buena) al no adaptarse correctamente al problema en cuestión.

FUNCIÓN DE EVALUACIÓN

Se encarga de juzgar cuan bueno es un individuo (o cuan buena es la solución que representa el individuo). Al igual que el genoma, es muy importante una correcta definición, pues puede llevar también a casos de sobreajuste, si la función es demasiado específica; o de “subajuste”, si es una función demasiado genérica, y el genoma no puede “adaptarse”.

Generalmente se suele tener una función de evaluación fija, que se mantiene constante a lo largo de toda la ejecución del algoritmo (distancia recorrida por un robot, altura máxima alcanzada en un cohete, distancia al objetivo...). Sin embargo, puesto que en realidad lo que intentamos es juzgar cuan buena es una solución, la función de evaluación puede ser, como en nuestro caso, hacer un combate entre los individuos de la población y ver cuál obtiene más victorias. De esta manera, tenemos una función de evaluación dinámica, en la que en una generación una criatura puede ser muy buena (ganar al resto de sus competidores), y unas cuantas generaciones más adelante ser una criatura de las peores (no ganando a nadie).

OPERADORES DE SELECCIÓN

Los operadores de selección son, básicamente, la manera en la que seleccionamos los individuos de cara a la continuación en la siguiente generación. Haciendo un símil con la naturaleza, mientras la función de evaluación determinaría cuánto de adaptada está o cuánto de capaz es una criatura es de estar en posición de reproducirse, el operador de selección vendría a ser la forma en que dos criaturas se ven “emparejadas” de cara a la reproducción (en el caso de criaturas con reproducción sexual). A lo largo de nuestra investigación utilizaremos dos tipos de selección: selección por ruleta y selección por torneo, cada una con sus ventajas e inconvenientes.

En la selección por ruleta a las criaturas se les asigna una probabilidad de ser elegidas para el cruce de acorde al valor obtenido en la función de evaluación. Cuanto mejor sea la criatura, más probabilidad tendrá de ser elegida. El problema de este tipo de selección es que una vez una criatura ha llegado a un máximo subóptimo y obtiene una gran probabilidad en la ruleta, se va a producir una convergencia muy fuerte a dicho individuo, negando la posibilidad de salir del mencionado máximo subóptimo.

La selección por torneo consiste en emparejar aleatoriamente las criaturas en un cuadro de eliminación, y en cada enfrentamiento quedarse con la criatura que tenga una mejor función de evaluación. Por ejemplo, con cuatro criaturas (A, B, C, D), se emparejaría aleatoriamente A con C y B con D, y nos quedamos con las que mejor valoración tengan (A, D). Entonces, estas dos criaturas son las que se acabaran cruzando para obtener el individuo de la siguiente generación. Lo positivo de este método es que da más oportunidades a los individuos con puntuaciones más bajas, aumentando la diversidad a costa de la velocidad de convergencia (si un individuo está en un máximo, le llevará más generaciones “imponer” su gen dominante).

OPERADORES DE CRUCE

Una vez hemos seleccionado dos criaturas para su “reproducción”, hay que definir como mezclamos los genes de ambas para obtener un nuevo individuo para la población siguiente. Esto depende en gran medida de la implementación del genoma de las criaturas, pero en general se pueden clasificar en los siguientes tipos:

- Cruce homogéneo: consiste en seleccionar aleatoriamente el gen de uno de los progenitores, y descartar el otro. Se usa principalmente con genes discretos, bien enteros, o bien caracteres o bien valores booleanos.
- Cruce por “media”: se obtiene la media (no necesariamente aritmética, ya que puede estar ajustada con un multiplicador aleatorio para evitar que se acabe comprimiendo al cero) de los genes de ambas criaturas.
- Cruce en un punto/en dos puntos/ en N puntos: Aunque hemos considerado que carece de sentido para nuestro caso concreto, este operador vendría a ser cortar la cadena de números que representan el genoma en $N+1$ trozos, y mezclar los trozos enteros, en lugar de gen a gen como en el homogéneo.

OPERADORES DE MUTACIÓN

Al igual que en la naturaleza pueden ocurrir mutaciones, tanto para bien como para mal, el algoritmo genético utiliza esta técnica para evitar que las soluciones representadas por los individuos caigan en un máximo relativo del espacio de búsqueda y no salgan de esa zona. Mediante la mutación, lo que conseguimos es “saltar” alejándose del máximo relativo en el que está atascada la solución, y llegar a otra solución que puede (o no) ser otro máximo relativo por el que diversificar la población.

Adaptación del problema al algoritmo genético

En este apartado hablaremos de cómo hemos adaptado el problema (ver que criaturas son más fuertes) a los conceptos de algoritmos genéticos. Para ello, vamos a describir el modelo de combate que tenemos, así como el modelo de criaturas y que características las diferencian unas de otras.

CRATURA

En primer lugar, cada criatura tendrá cinco características físicas: aguante (HP), ataque (AT), armadura (AR), energía (EN) y agilidad (AG). La suma de todas ellas tiene que ser 30, para que de esta forma las criaturas estén equilibradas.

- HP: Determina la cantidad de puntos de vida con la que empiezan los combates.
- AT: La cantidad de puntos de daño que potencialmente pueden producir las criaturas con su ataque.
- AR: Proporcional a los puntos de daño absorbidos en cada ataque recibido.
- EN: Recurso necesario para poder realizar acciones en el combate; ya sea atacar, defenderse... En la sección de combate se explicará más detalladamente su uso.
- AG: En el caso de que la criatura opte por intentar esquivar un ataque, la agilidad marcará la probabilidad de que esto ocurra.

Además de las características físicas, las criaturas dispondrán también de un sistema que les permite, a partir de una serie de indicadores durante el combate (diferencia en puntos de vida actuales, diferencias en ataque...) tomar la decisión de que acción realizar en cada turno. Cada par acción-indicador tendrá una ponderación, y se tomará la acción con la suma de indicadores ponderados más alta.

GENOMA

Por lo tanto, el genoma lo representaremos como una lista de números, de los cuales los 30 primeros serán enteros representando la cualidad que mejoran (ejemplo más abajo), y los siguientes 20 serán las ponderaciones de las acciones a realizar.

COMBATE

Por otro lado, el combate entre dos criaturas funciona como sigue:

- Cada criatura empieza el combate con la cantidad de puntos de vida que tenga como HP.
- Los turnos son simultáneos. Es decir, las dos criaturas tienen que tomar la decisión sobre qué acción realizar sin saber que va a hacer el otro.

- Existen cuatro posibles acciones en cada turno: Atacar, Bloquear, Esquivar o Pasar.
 - Atacar necesita dos puntos de energía, e intenta causar daño al rival; con el máximo sienta el valor de AT.
 - Bloquear necesita tres puntos de energía, y asegura que independientemente de lo que haga el rival no recibe puntos de daño.
 - Esquivar intenta evitar el daño del rival. Necesita un punto de energía, en contraposición a bloquear que necesita 3. Sin embargo, la probabilidad de evitar el daño oscila entre 0.33 y 0.66, de acorde con la cantidad de Agilidad que tenga la criatura.
 - Pasar regenera un punto de energía, pero deja vulnerable a la criatura al daño.
- El combate acaba cuando una o ambas criaturas están con cero puntos de vida o menos. Entonces, la criatura con menos puntos de vida (incluyendo números negativos), pierde. En el caso de que ambas criaturas acaben el combate con los mismos puntos de vida se produce un empate.
- De cada combate se guarda la información no solo de quien ha salido victorioso, sino también del daño causado, recibido, y la diferencia de puntos de vida entre las dos criaturas.

FUNCIÓN DE EVALUACIÓN

En la función de evaluación no solo se tiene en cuenta el número de victorias de la criatura, sino también el daño total que ha causado y la suma de diferencias de vida al final de los combates. De acorde a estos valores, se le otorga una puntuación mediante esta fórmula:

$$10 \cdot \text{NúmeroVictorias} + 2 \cdot \text{DañoHecho} + \text{DiferenciaVida}$$

Planteamiento Inicial

Inicialmente decidimos empezar con un ADN formado por una lista de 30 números enteros representando cada uno la cualidad que mejoraban, y 20 números en coma flotante, representando las ponderaciones de las acciones a realizar, y comprendidos en el intervalo $(-10,10)$. Al principio parecía la opción más lógica teniendo en cuenta lo que representaba cada número. Además, descartamos la opción de tener cinco números representando la cantidad de cada característica, porque entonces complicaría mucho más las operaciones de cruce (al tener que descartar todo individuo que no tuviera una suma de 30). En realidad, el hecho de descartar individuos generados a partir del cruce no es algo “prohibido” en los algoritmos evolutivos (vendrían a ser en la naturaleza las criaturas con defectos genéticos que mueren prematuramente), pero sí es cierto que era complicar en sobremanera la operación de cruce y mutación.

Como operador de selección escogimos la ruleta, ya que era el más simple de implementar, y para una primera aproximación sobre la que ir observando resultados era válido. Esto supondría que, como se explica en la introducción de los conceptos de algoritmos genéticos, las criaturas más dominantes en las primeras generaciones arrastraran a toda la población a ese máximo relativo.

Para el cruce escogimos cruce uniforme (seleccionar uno de los parientes) en el caso de los genes relacionados con características físicas, y una media aritmética simple para el caso de los genes que moderaban el comportamiento, lo cual nos llevaría a un problema de convergencia al cero.

La mutación venía dada para cada gen de manera individual (es decir, cada gen tenía una probabilidad de mutar de manera independiente al resto de genes del individuo), dada por un ratio inicialmente en un 3%, pero restringido únicamente a los genes de comportamiento, lo que nos llevaría a otro problema de diversidad con las características físicas. En el caso que se diera el caso de mutar, el gen podía moverse en un rango de $(-1.5,1.5)$ con respecto a su valor original, siempre respetando los rangos de $(-10,10)$.

Problemas y soluciones

Los primeros problemas que tuvimos con el planteamiento inicial estaban relacionados con la diversidad de los individuos y con la manera en la que los seleccionábamos y cruzábamos en las generaciones siguientes. El hecho de utilizar ruleta significó que los individuos que obtenían alguna mínima ventaja (lo cual era fácil teniendo en cuenta que al principio todas las puntuaciones eran muy bajas), adquiriría una gran porción de la ruleta, y por ende acababa repetido muchas veces en la generación siguiente.

Con el paso de las generaciones, esto provocó un fallo en la función de evaluación (el combate). Pues al ser iguales todos los combates acababan en empate, con el mismo daño hecho por uno y por otro. Esto es, todos los individuos terminaban siendo iguales y con el mismo valor en la función de evaluación.

Como primera medida para remediar esto cambiamos el operador de selección de ruleta a torneo, para que los individuos con una puntuación considerablemente mayor no acapararan toda la siguiente generación. Este cambio resultó tal y como se esperaba a corto plazo, pero a la larga (al cabo de unas cuarenta o cincuenta generaciones, a diferencia de con ruleta, en la que la población convergía con tan solo cinco o seis rondas) acababa ocurriendo lo mismo.

En este punto, consideramos dos caminos a seguir: por un lado forzar mediante el análisis estadístico en la población a que las criaturas fueran diferentes (posteriormente me adentraré más en este caso), o implantar lo que nosotros llamaremos “población de prueba” o “población de test”, que será la población a la que enfrentaremos los individuos en lugar de contra sí mismos.

De cara a poder analizar el efecto que cada una de ellas tenía por separado, decidimos implementar primero la población de prueba, pues nos resultaba más fácil en cuanto a programación ser refiere, y preferíamos poder analizar nuevos resultados cuanto antes.

La población de prueba era una población del mismo tamaño que la población normal (que llamaremos para diferenciarla “de evolución”), y en esta primera aproximación se generaba aleatoriamente al principio de la ejecución del programa, manteniéndose durante todo el proceso. Pese a que el problema de convergencia y empate se eliminó, surgió otro problema causado por la implementación de la población de prueba. Al mantenerse durante todo el proceso de evolución la misma población de prueba, las criaturas de la población de evolución conseguían llegar al máximo de puntuación posible, y a partir de ahí no cambiaban apenas. Dicho de otro modo, se estaba entrenando a las criaturas para combatir contra *esa* población de prueba en concreto, pero no contra criaturas en general. Es decir, si cambiabas la población de prueba después de haber conseguido una convergencia las mismas criaturas que ganaban con la puntuación máxima ahora no conseguían puntos.

Para evitar esto, la solución era tan simple como ir sustituyendo las criaturas de la población de prueba por criaturas mejores (obtenidas mediante la población de evolución). Además, y aprovechando la otra rama de mejora que nos habíamos dejado antes (análisis estadístico de la población con respecto a sus características físicas), incorporamos una

función que nos decía cuanta diferencia había entre cada una de las características físicas entre las dos. Esta función era el primer paso para incorporar funciones de media y desviación estándar con respecto a la media de la población, esta última aportándonos ya por fin una medida adimensional de cuando de diferente era una criatura respecto a la media.

El proceso ahora consistía en sustituir, después del proceso de creación de la siguiente generación evolutiva, las tres criaturas de la población de prueba más cercanas a las tres mejores criaturas de la población evolutiva. Es decir, obteníamos las tres mejores criaturas de la última generación, y cambiábamos la criatura de la población de prueba que más se asemejara a cada una de ellas por dicha criatura. De esta manera, la población de prueba ahora consistía en un grupo de criaturas diversas (porque sustituíamos únicamente la más próxima) y medianamente poderosas (porque eran las mejores de las generaciones anteriores). El problema de que todas las criaturas obtenían la puntuación máxima había sido corregido con esto.

Sin embargo, no estábamos del todo contentos con la diversidad de la población de evolución, pues, aunque no solía haber más de dos o tres individuos iguales (con las mismas características físicas) de una población de veinte, las diferencias eran de un punto arriba en una cualidad y un punto abajo en otra. Por este motivo, decidimos forzar aún más la diversidad penalizando y bonificando a los individuos según lo parecidos que eran a la media. Su puntuación era multiplicada por la desviación estándar, y de esta manera individuos que fueran muy parecidos a la media, pero sin una puntuación mejor que sus vecinos, acababan teniendo pocas probabilidades de sobrevivir en la siguiente generación.

Aun así, todavía no se conseguía la diversidad perseguida, pues los individuos seguían diferenciándose de, como mucho, dos o tres puntos en alguna característica. Por ello, y como último paso antes de concluir la investigación en el proyecto, decidimos introducir un enfoque por islas, que evolucionaban de manera independiente unas de otras. De esta forma, mediante una población de prueba global (en todas las islas se utilizaba la misma), cada isla tomó un rumbo evolutivo diferente, llegando a resultados finales muy distintos.

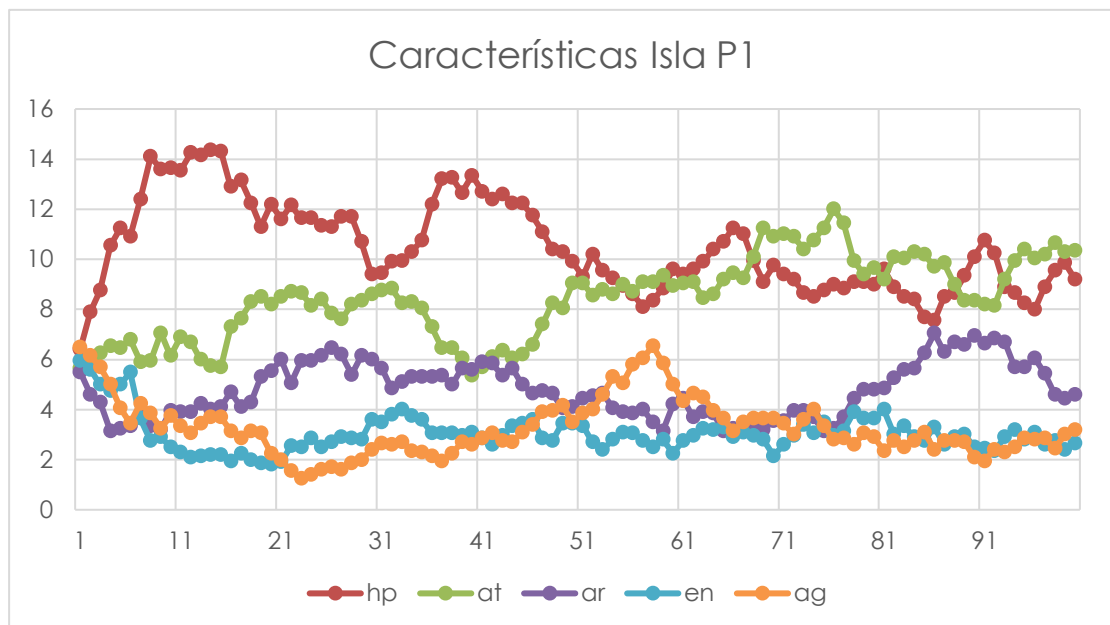
Cada diez generaciones se hacen cinco intercambios de individuos al azar de entre todas las islas. No se deben intercambiar muchos individuos entre las islas, ya que de ser así perderíamos la independencia entre las islas que buscábamos, cayendo así en la formación de una sola gran isla. Por otro lado, si la cantidad de intercambios es menor, no se produce ninguna relación entre las islas, y por lo tanto en realidad es como si tuviéramos una cantidad de ejecuciones independientes del algoritmo. La idea es un punto intermedio, que permita de manera independiente a las islas evolucionar, y si esa evolución supone una mejora sustancial, que el cambio se contagie al resto de islas.

El número de generaciones que marcan el ritmo de intercambio entre las islas del número de generaciones que tardaba el algoritmo en converger a algo estable. En nuestro caso, como no superaban las 100, decidimos que un buen número podría ser cada diez generaciones. Así, las islas tenían tiempo de asimilar los cambios e introducirlos con el genoma particular que tuviera esa isla en concreto.

Resultados finales

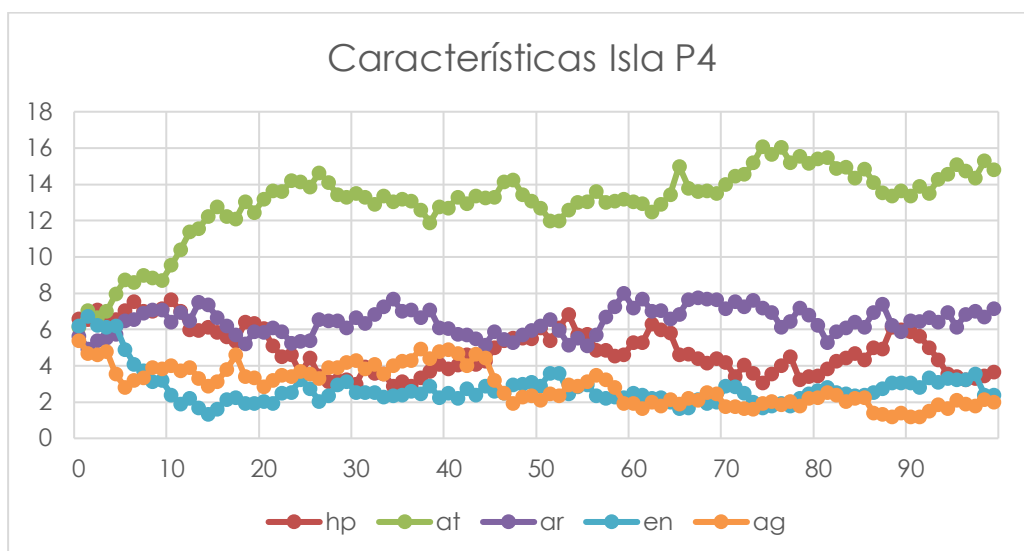
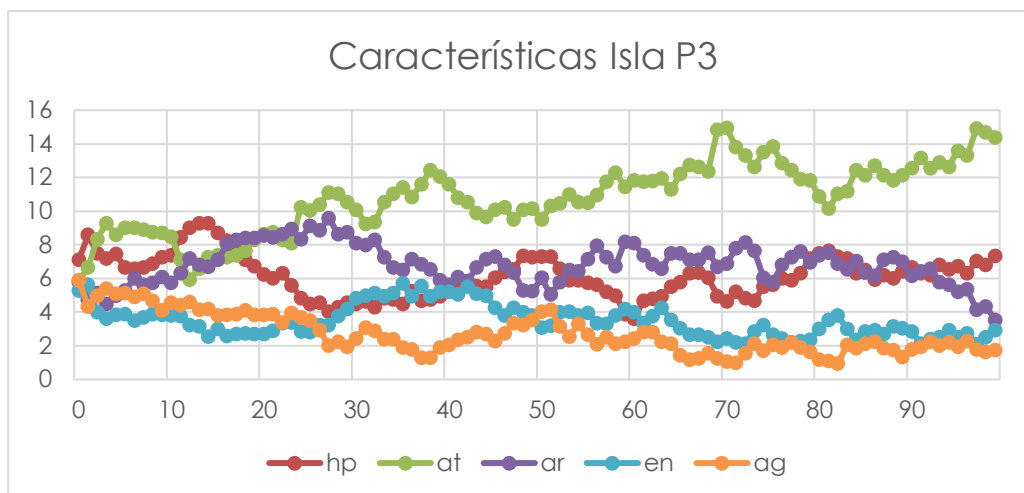
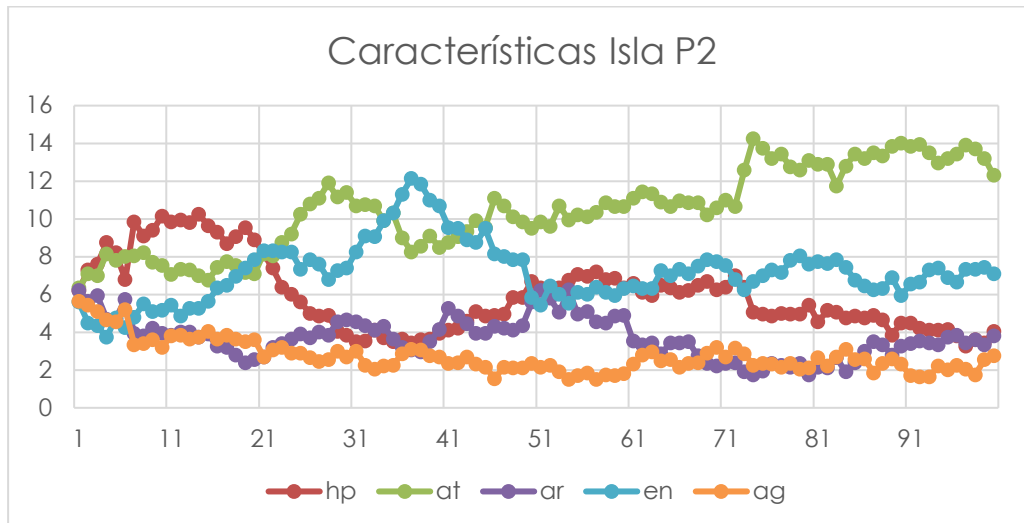
Ahora vamos a analizar y exponer el resultado final tras una ejecución, pues como se expuso anteriormente, cada vez se crean una población inicial diferente. Para ello, nos vamos a ayudar de los gráficos que hemos creado con los datos obtenidos durante la ejecución.

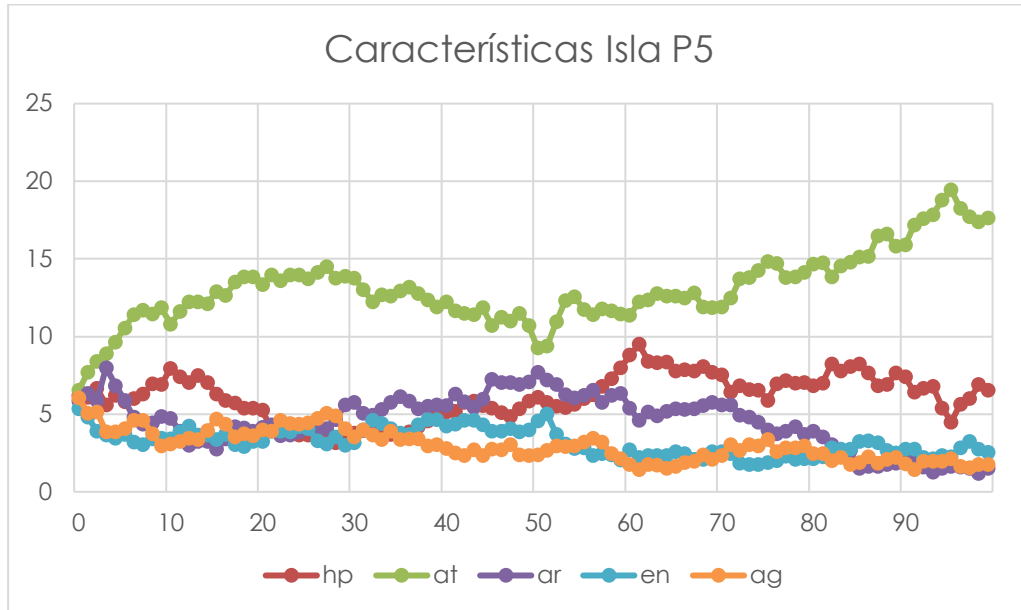
Primero analizaremos las características, para ello, creamos una gráfica representando con un color diferente cada una de ellas. Obteniendo así, la evolución con respecto al paso de las generaciones de todas las islas.



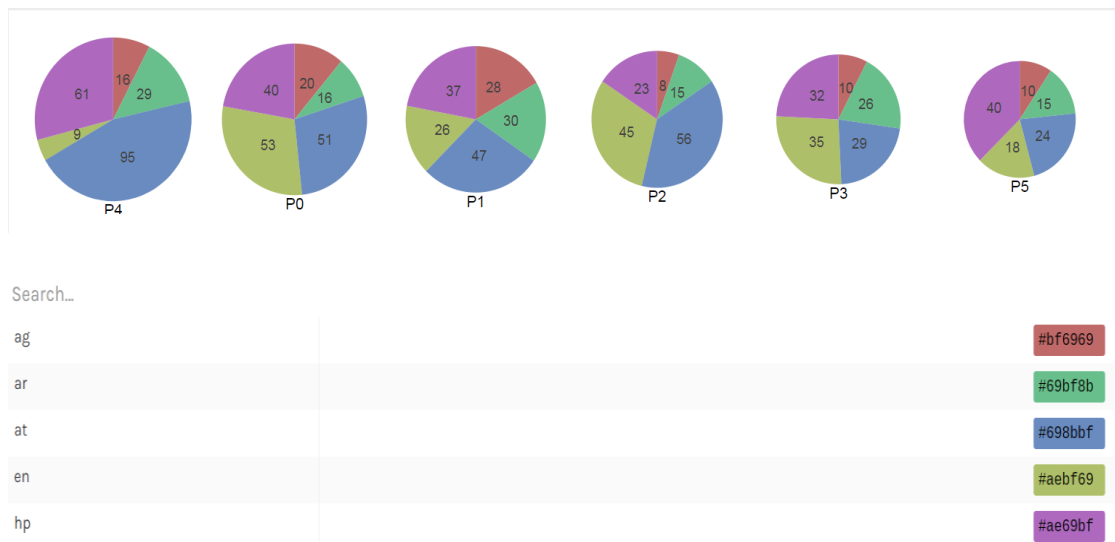
Si nos fijamos, podemos percatarnos que la mayoría de las características no llegan a converger claramente. Para nosotros esto no es un problema, debido a que éstas estarán fluctuando por las medidas con las que hemos asegurado la diversidad; como por ejemplo el intercambio de criaturas por el mecanismo de islas. Es más, se puede ver que en las generaciones número 10, 20, 30... se suelen producir cambios de tendencia en la evolución, provocados por el intercambio de individuos entre las diferentes islas.

Ponemos el gráfico de todas las islas restantes (porque han sido sólo cinco en nuestro caso):





Ahora analicemos las islas en su conjunto, viendo claramente la imagen global.



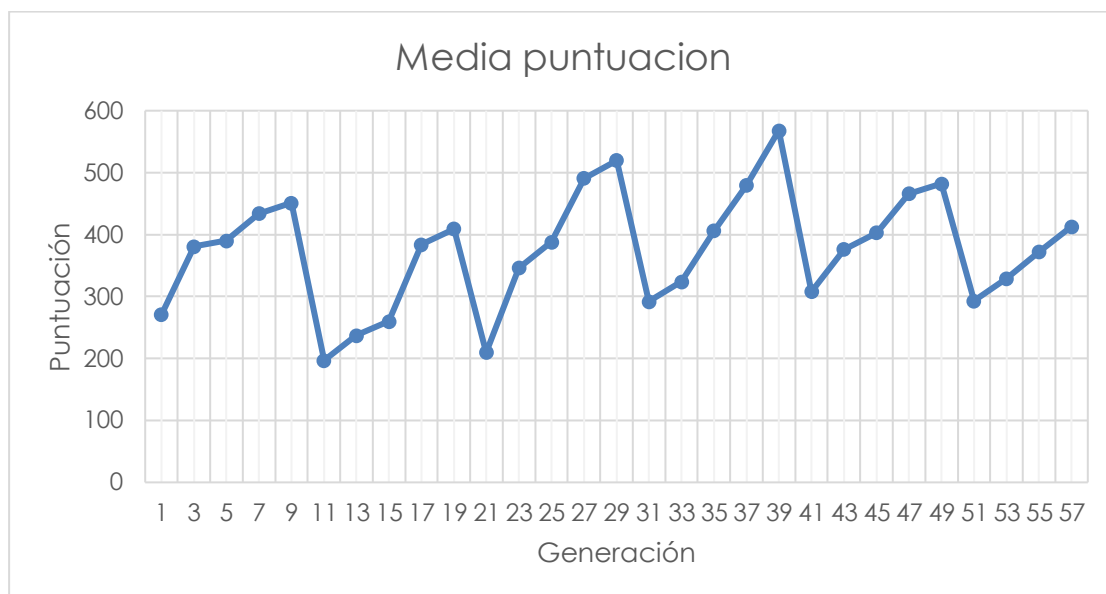
Con este último gráfico podemos sacar las siguientes conclusiones:

Como en todas las últimas generaciones, el HP y el AT han aumentado considerablemente con respecto a las otras características. Una buena hipótesis podría ser que el HP y el AT son indispensables a la hora de hacer que la criatura sea mejor que las otras. Es decir, que sin un alto valor de estas características, es bastante probable que la criatura no gane combates y, en consecuencia, no consiga tener descendencia. Aunque si hemos de ser exactos, hay islas como la P2 o la P3, cuyos altos niveles de EN superan con creces a las

características mencionadas; podrían considerarse ejemplos que niegan la hipótesis planteada. Esto también demuestra empíricamente que las características que nosotros podamos pensar en un primer momento que son las más importantes no tienen por qué ser las que se obtienen mediante la búsqueda por evolución. De todas formas hemos de tener en cuenta que, como la población original es diferente en cada primera ejecución, no tiene por qué ser necesario que todo lo mencionado sea cierto para otra ejecución diferente.

Por otro lado, queda perfectamente reflejado el haber cumplido con nuestro objetivo de la diversidad. Si tenemos en cuenta los datos, podemos extrapolar que la “criatura media” (la criatura con la media de las características entre todas las islas), que representaremos como la “isla 0” o P0. Solo nos hace falta darnos cuenta de este detalle, de ver que no es homogénea la distribución de las características en las islas, para estar satisfechos con nuestros resultados. Se puede ver a simple vista que cada isla ha engendrado a criaturas muy dispares, aunque en realidad al existir una conexión entre las islas (intercambio de individuos), se trata de una evolución independiente influenciada por los resultados en otras islas.

Ahora, vamos a analizar y comprobar que efectivamente las criaturas que se han ido formando son, como esperábamos, mejores que las anteriores. Para ello lo que vamos a utilizar es el siguiente gráfico:



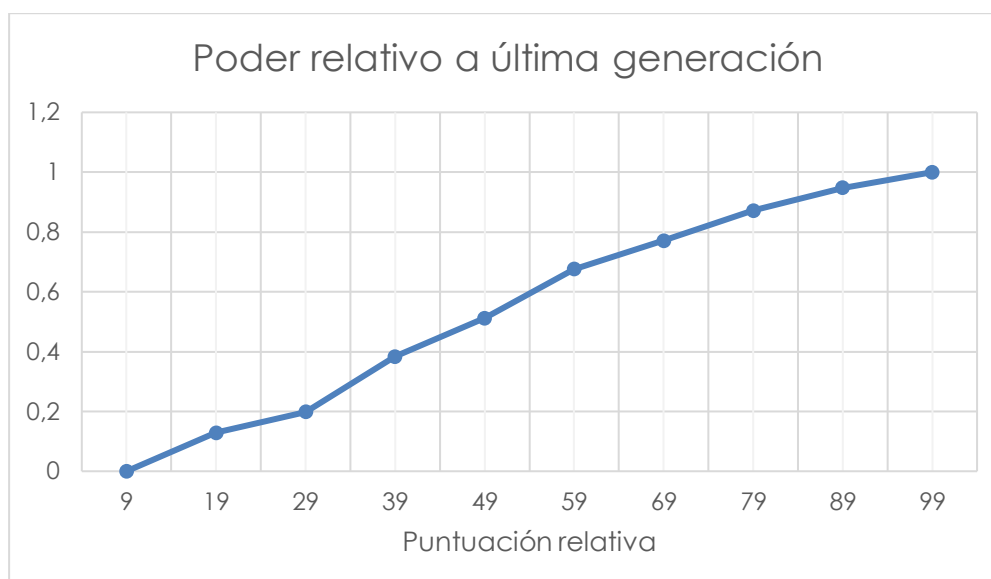
El gráfico consiste en las puntuaciones medias (entre todas las islas) obtenidas mediante la función de evaluación. Se puede ver como al principio de cada diez generaciones (gen. 1, 11, 21...) la puntuación baja drásticamente, ya que es cuando se actualiza la población de prueba con los mejores individuos de entre todas las islas. Durante las generaciones intermedias las criaturas de las islas van evolucionando y mejorando sus resultados contra esa población de prueba, hasta que llega la nueva generación de prueba que limita los resultados.

Cabe destacar también que el cambio de los individuos de isla se produce en la generación siguiente al cambio de la población de prueba, por lo que las alteraciones que surgen del intercambio de individuos se van fraguando durante las diez generaciones que dura el ciclo de esa población de prueba.

Sin embargo, esto no nos asegura que, por ejemplo, las criaturas de la decimoséptima generación sean peores o mejores que las de la última. La mejor manera para ver si de verdad ha habido una progresión en la calidad de las criaturas es guardar las diferentes poblaciones de prueba que vamos obteniendo, y enfrentar a la última generación evolutiva que obtenemos a cada una de las poblaciones de prueba. Si efectivamente ha habido mejora, la puntuación al enfrentarse a la última población de prueba va a ser mejor que al enfrentarse a la primera generación de prueba.

Para ponerlo en forma de símil, un alumno que cursa cuatro convocatorias de una asignatura se considera que ha mejorado si, tenido en cuenta que los exámenes se van haciendo más complicados, adaptándose al alumno para que su nota sea lo más próxima al cinco, al enfrentarle a todos los exámenes después del cuarto año la nota del primero es altísima (porque en ese momento su conocimiento era bajo y el examen al adaptarse al cinco tenía dificultad baja), y la nota del primero es próxima al cinco (el ultimo examen está adaptado a su conocimiento actual, que es ya muy alto).

Por último, si consideramos el poder (la puntuación) de la ultima generación con respecto a la primera población de prueba como un cero, y el poder de la ultima generación con respecto a la última población de prueba como un uno, y reajustamos linealmente las notas intermedias, obtenemos el siguiente poder relativo:



Y como se puede observar, efectivamente se cumplen nuestras expectativas. Vemos que la progresión de las puntuaciones ha sido ascendente, pues cada vez las criaturas van mejorando; y lineal, pues el progreso es continuo de generación en generación. Esto

significa que, si sometemos a la población de la generación número cincuenta y la de la generación número cien a un combate contra una misma población, aproximadamente la de la generación número cien sacara el doble de puntuación que la de la generación número cincuenta.

Conclusiones y Trabajo Futuro

OBJETIVOS ALCANZADOS

Los datos corroboran que hemos cumplido con nuestro principal objetivo. La simulación al proceso de optimización que se da en la naturaleza, esto es, la evolución; ha sido en gran medida replicada con gran fidelidad. Los datos confirman que las criaturas han ido progresando cualitativamente. Además, hemos alcanzado el objetivo de la diversidad, uno de los objetivos más importantes que más ha costado durante todo este proyecto. Aparte de simular con mayor precisión el proceso evolutivo de la naturaleza, también, en términos computacionales, hemos conseguido la forma alcanzar máximos locales. En nuestro planteamiento, cumplimos este objetivo al generar criaturas muy poderosas y a la vez muy diferentes.

OBJETIVOS NO ALCANZADOS

Hay cosas que no hemos llegado a implementar dado a que aumentaban de gran manera la complejidad. Como, por ejemplo, la simultaneidad a la hora de perecer o nacer criaturas nuevas. O, por otra parte, el de descartar individuos generados a partir del cruce, por razones similares.

POSIBLES MEJORAS

En primer lugar, tendríamos que resaltar todo lo que no hemos mencionado que no hemos llegado a implementar. Por otro lado, aunque la parte de la toma de decisiones de las criaturas apenas la hemos tenido en cuenta al final, se puede mejorar para que tenga una mayor relevancia en los resultados de los combates.

POSIBLES APLICACIONES

Aun siendo el problema planteado muy concreto, podemos llegar a posibles utilidades, siempre y cuando la implementación del problema sea correcta.

Todo videojuego multijugador online, en concreto los llamados PvP (Player vs Player), se enfrentan a un problema común a la hora de crear o diseñar el juego. Esto es, el balanceo. El balanceo se refiere al equilibrio que tiene que estar presente en las cualidades de los personajes jugables; o dicho de otra forma, a la habilidad que tiene que tener el diseñador del diseñador del videojuego a la hora de crear los personajes jugables o hasta incluso el entorno. Pues si haces uno más fuerte que el resto, todos los jugadores acabarán por usar ese y el juego dejará de ser entretenido. De la misma forma, también se tiene que poner atención a la diversidad de personajes, claro está que si todos son iguales, el videojuego

deja de ser entretenido. Es una balanza de fuerzas constante en los cuales que, siempre que se quiere meter un nuevo personaje, uno se tiene que rascar la cabeza porque aumenta el problema. Este problema de actualidad lo sufren y tratan de salir del paso gigantes de la industria del videojuego como *Riot Games* con su videojuego *League of Legends* o Blizzard con *Overwatch*. Cabe destacar que la propia comunidad de jugadores es al menos consciente de estos problemas y no pocas veces con agrado.

Por esto y todo lo dicho, creemos que con este proyecto sentamos las bases para poder tratar de formas más efectiva estos problemas antes mencionados. Se podría dejar jugar a los personajes jugables entre sí en millones de combates y miles de generaciones, para que cuando llegue el producto final al cliente, tenga un juego justo y balanceado. Ya que se habrían logrado crear de antemano personajes capaces de jugar entre ellos de manera que sea entretenida e interesante y, que sean todos diferentes.

CONCLUSIÓN FINAL

Como conclusión podemos comentar que hemos aprendido en qué consisten, la idea intrínseca, de los algoritmos genéticos. Sus capacidades y sus limitaciones. Lo importante que es un buen planteamiento de ellos, ya que como hemos aprendido, depende lo que evalúes, a la hora de saber si los resultados son satisfactorios o no. Con ellos hemos estudiado en profundidad una poderosa herramienta que nos ayuda a dar con las mejores soluciones aproximadas a problemas muy complejos. Y también cabe mencionar, que hemos adquirido una comprensión muy práctica tanto de los conceptos de la evolución biológica y hasta las ideas básicas de la evolución genéticas. También hemos aprendido que llevar una correcta cronología del proyecto ayuda a la hora de poder trazar mejor el proceso llevado a cabo, pues permite, en caso de no saber que rumbo tomar, haces retrospectiva y poder analizar con más detalle los posibles caminos a seguir.