



(54.623) Juan Dantur (55.382) Ariel Debrouvier (53.396) Agustín Golmar



Tabla de Contenidos

Implementación	2
Introducción	2
Arquitectura	2
Cromosoma	2
Configuración Óptima	3
Adaptación	3
Selección/Reemplazo	3
Cruce	3
Mutación	3
Resultados	4
Anexo	6
Figuras	6
Demostraciones	7
Optimización ATM/DEM	7
Adaptación Máxima	8
Benchmarks	o



1. Implementación

A continuación, se detallarán las características *funcionales* y *no-funcionales* del sistema desarrollado, junto con las decisiones de diseño tomadas.

1.1. Introducción

Se implementó un RCGA (Real-Coded Genetic Algorithm), para optimizar la configuración de ítems en un videojuego de roles, maximizando el desempeño de la clase asignada (**Defensor N° 2**). En las siguientes secciones se detallarán las decisiones tomadas en cada etapa del desarrollo.

1.2. Arquitectura

Se implementó en Matlab y al igual que el trabajo desarrollado en el *TP N° 2: Redes Neuronales*, se utilizó la librería *JSONLab 1.5* para deserializar el archivo de configuración. Además se implementó un sistema de gráficos en tiempo real, lo que permite visualizar las variaciones de adaptación conforme el sistema se ejecuta.

Debido a que la base de datos es considerablemente mayor a la que se utilizó en el *TP N° 2*, se hizo especial hincapié en la performance y en la carga de datos. Se diseñó un sistema de carga *lazy*, el cual permite modificar la configuración de parámetros sin reinstalar la base de datos.

1.3. Cromosoma

El cromosoma está compuesto por una tupla de 6 elementos:

(Armor, Boots, Gauntlet, Helmet, Weapon, Height)

Donde los primeros 5 elementos representan el número de fila en la base de datos correspondiente (i.e., un número natural), y el último gen es un número real en el intervalo [1.3, 2.0].

Luego de analizar las ecuaciones de desempeño por clase, los *stats* por ítem, y el cálculo final de ataque y defensa, se determinó que la única variabilidad compleja es introducida por los modificadores de ataque y defensa en función de la altura del personaje (*Figura* 1). Esto se debe a

que los stats finales se encuentran acotados (Sección 3.2.2), y además su función es creciente en el dominio de los datos disponibles. Si el algoritmo fuera capaz de diseñar un cromosoma específico (variando los stats de cada ítem), tendría una tendencia a incrementar la magnitud de los genes hasta saturar las tangentes hiperbólicas, por lo cual la única restricción es optimizar el modificador de ataque y defensa, variando la altura en el rango adecuado. Ya que la configuración final de ítems debe existir en la base de datos, no es posible modificar libremente los stats.

1.4. Configuración Óptima

Se explicará etapa por etapa la forma en la que se seleccionó la mejor configuración para resolver el problema propuesto, junto con las decisiones de diseño.

1.4.1. Adaptación

Para acelerar el cálculo de la adaptación se aplicó la *regla de Horner* durante el cálculo de los modificadores (*ATM* y *DEM*), así como se observó que los modificadores dependen de un mismo factor en común (ver las ecuaciones en la *Figura 2*).

Otra optimización consiste en no computar la adaptación sobre toda la población en cada ciclo, sino únicamente en los individuos modificados. De todas formas, la adaptación relativa y acumulada deben re-computarse cada vez.

Luego de realizar una estimación analítica (Sección 3.2.2), se determinó que la máxima adaptación posible (dada por la altura óptima h = 1.3), no puede superar los 246 puntos, y utilizando los datos provistos por la cátedra, los 176 puntos.

1.4.2. Selección/Reemplazo

Se ejecutaron pruebas aisladas sobre cada método de selección implementado (boltzmann, elite, torneo probabilístico y determinístico, ranking, ruleta y universal), dejando el resto de los parámetros fijos y se eligieron los métodos que tras 5000 generaciones tuvieron el mejor fitness. De igual forma se trabajó sobre los métodos de reemplazo.

1.4.3. Cruce

Utilizando los dos métodos elegidos en la etapa anterior, se realizaron pruebas sobre los 4 tipos de cruza (un punto, dos puntos, anular y uniforme) para determinar cuál es el mejor.

1.4.4. Mutación

Se realizaron pruebas para distintos valores de probabilidad de mutación dejando fijos los otros parámetros, con el objetivo de determinar qué probabilidad de mutación brindaba una mejor solución.



Resultados

A partir de las pruebas aisladas realizadas con cada método de selección, se eligieron los dos métodos que tuvieron mejor desempeño a lo largo de 5000 generaciones. Los mismos fueron Élite y Torneo Determinístico. Se eligieron 2 ya que la implementación de los algoritmos permite la hibridación de ambos y se decidió darle un porcentaje de 0.7 a Torneo Determinístico y de 0.3 a Élite, dándole así menor influencia a este último método ya que puede provocar el estancamiento en un máximo local. En la *Tabla 1* del Anexo, se observa el desempeño final de cada método. Para el método de Boltzmann, se optó por utilizar una temperatura inicial de 100 y una tasa de reducción de 0.001, ya que con una temperatura inicial mayor la presión de selección era muy baja, lo que generaba amplia oscilación en el fitness máximo. Por otra parte, con una tasa de reducción mayor la temperatura descendía muy rápidamente, lo que causaba una convergencia prematura.

Tras fijar los métodos de selección y efectuarse las pruebas sobre las técnicas de cruza, no hubo ninguna técnica que ofrezca un desempeño superlativo con respecto a las otras. Se eligió la técnica anular ya que si bien no tuvo mucho ventaja, fue la que consiguió un mejor desempeño (ver *Tabla 2*).

Para los métodos de reemplazo, el método 1, que sustituye a toda la población (G = 1), tuvo mejor desempeño que los métodos 2 y 3 que conservan a gran parte de la generación anterior (*Tabla 3*). Este método tiene buenos resultados, siempre y cuando la selección de padres no sea completamente elitista y descarte padres que en alguna cruza brinden un mejor desempeño.

En las pruebas para determinar el tamaño de la población, a medida que se aumentó el mismo, el mejor fitness incrementó, pero al mismo tiempo el tiempo de procesamiento requerido (Tabla 4). Para el problema en cuestión al tener 5 genes con 10 millones de posibilidades cada uno y una altura que es un número real, son mejores aquellas poblaciones que tienen un número elevado de individuos. Sin embargo, esto conlleva a un uso intensivo de los recursos de la máquina y un mayor tiempo de ejecución por cada generación.

Para las pruebas restantes se decidió mantener una población de 300 individuos, ya que este tamaño permite ejecutar con mayor rapidez los distintos casos de prueba, con los recursos que poseía el grupo para ejecutar el proyecto.

En el gráfico correspondiente a la configuración óptima puede verse que el fitness medio siempre cambia de generación en generación, por lo que se decidió no analizar el corte por estructura, ya que ésta siempre varía significativamente debido a la probabilidad de mutación. Además se puede notar que el fitness máximo se estabiliza

y deja de crecer en las últimas 3000 generaciones, por lo que se concluye que un parámetro adecuado para el corte por contenido sería alrededor de 3000 generaciones seguidas sin crecimiento (*Figura 3*).

Otro parámetro que se tuvo en consideración fue la cantidad k de padres elegidos para cruzar y generar la próxima generación, el mismo también determina la cantidad de hijo que se reemplazarán (*Tabla 5*). Para la población de 300, un valor de 50 consiguió un mejor fitness promedio final.

Finalmente, al evaluar la probabilidad de mutación se pudo notar que un valor de 0.07 brindaba el máximo fitness respecto de los otros valores evaluados (*Tabla 6*).

3. Anexo

En esta sección se despliegan las figuras referenciadas, las demostraciones analíticas de algunos parámetros, y los resultados numéricos obtenidos durante la etapa de testing y benchmarking.

3.1. Figuras

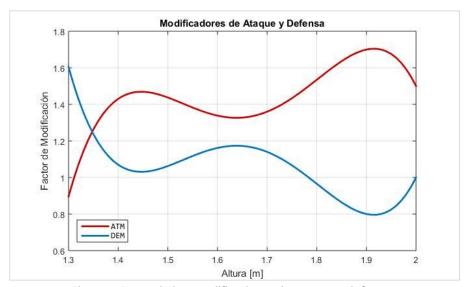


Figura 1: Curva de los modificadores de ataque y defensa.

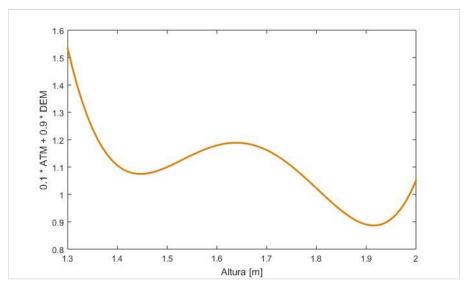


Figura 2: Variación parcial del desempeño final, en función de la altura.

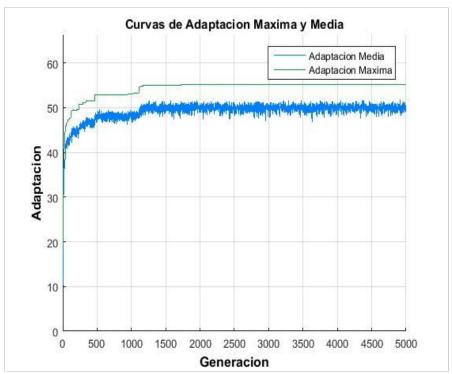


Figura 3: Adaptaciones media y máxima con parámetros óptimos.

3.2. Demostraciones

3.2.1. Optimización ATM/DEM

Los modificadores de ataque y defensa se definen como:

$$ATM(h) = \frac{1}{2} - (3h - 5)^4 + (3h - 5)^2 + \frac{h}{2}$$

$$DEM(h) = 2 + (3h - 5)^4 - (3h - 5)^2 - \frac{h}{2}$$

Se puede aplicar el siguiente reemplazo:

$$\nabla(\mathbf{h}) = (3h - 5)^2$$

Y sobre él, la regla de Horner:

$$\nabla$$
(h) = $h(9h - 30) + 25$

Ahora se define un polinomio en común, entre los dos modificadores:

$$\Delta(h) = \nabla(h)^2 - \nabla(h) - \frac{h}{2}$$

Finalmente, se redefinen los modificadores:

$$ATM(h) = \frac{1}{2} - \Delta(h)$$

$$DEM(h) = 2 + \Delta(h)$$

3.2.2. Adaptación Máxima

Los stats finales vienen definidos por:

$$\begin{split} Fuerza_p &= 100 \tanh \left(0.01 \times M_{fuerza} \sum Fuerza_{item}\right) \\ Agilidad_p &= \tanh \left(0.01 \times M_{agilidad} \sum Agilidad_{item}\right) \\ Pericia_p &= 0.6 \tanh \left(0.01 \times M_{pericia} \sum Pericia_{item}\right) \\ Resistencia_p &= \tanh \left(0.01 \times M_{resistencia} \sum Resistencia_{item}\right) \\ Vida_p &= 100 \tanh \left(0.01 \times M_{vida} \sum Vida_{item}\right) \end{split}$$

Donde *M* representa los multiplicadores por clase. Ahora bien, la tangente hiperbólica está acotada superiormente por 1. De esta forma (y sin importar el multiplicador, o la configuración de ítems), se tiene:

$$Fuerza_p, Vida_p \leq 100$$

$$Agilidad_p, Resistencia_p \leq 1$$

$$Pericia_p \leq 0.6$$

De igual forma, el ataque y la defensa se acotan:

$$Ataque \le 160 \times ATM(h)$$

 $Defensa \le 160 \times DEM(h)$

Por último, el desempeño final es:

$$Desempe\tilde{n}o \leq 160(p_{ataque} ATM(h) + p_{defensa} DEM(h))$$

Al observar la *Figura 2* (resultado de graficar este desempeño en el intervalo [1.3, 2]), se determina que la altura óptima es h = 1.3, y que el desempeño queda acotado superiormente por 246 (para un ATM = 0.8959, DEM = 1.6042, p_{ataque} = 0.1 y $p_{defensa}$ = 0.9).

Es posible computar un límite máximo inferior, utilizando los máximos *stats* para cada ítem en la base de datos provista por la cátedra. En este caso, se computa una matriz de *5x5*, donde cada fila representa un ítem, y cada columna el máximo valor hallado en el set de datos. Esto permite obtener una estimación de 176 puntos.

Sin embargo, luego de computar las diferencias entre estos vectores ideales con respecto a los que se encuentran en la base de datos, se determinó que aún así, el límite de 176 sigue siendo considerablemente más grande al que el algoritmo genético podría alcanzar luego de sucesivas generaciones y una configuración adecuada.

3.3. Benchmarks

Selection Method	Fitness
Boltzmann¹	50.938
Deterministic Tournament	51.663
Elite	51.530
Probabilistic Tournament	49.786
Ranking	50.648
Roulette	44.942
Universal	45.096

Tabla 1: Pruebas aisladas para el método de selección. Se utilizó una población de 300 individuos, k = 150 y 5000 generaciones. En verde, los métodos elegidos.

9

¹ Para el método de Boltzmann se utilizó una temperatura inicial de 100 y se redujo en un 0.1% la misma en cada generación.

Crossover Method	Fitness
Anular	52.945
Single-point	51.847
Two-point	52.378
Uniform	52.692

Tabla 2: Las adaptaciones alcanzadas variando el método de cruce, bajo los métodos de selección determinados por la *Tabla 1*.

Replacement Method²	Fitness
Method 1 (G = 1)	54.099
Method 2 (G = k/N)	52.456
Method 3 (0 \leq G \leq k/N)	50.128

Tabla 3: Pruebas sobre los métodos de reemplazo, utilizando nuevamente las heurísticas óptimas de la Tabla 1-2.

Población (N)	Fitness	Tiempo [s]
100	47.114	251.770
300³	51.583	725.560
600	54.233	1290.300

Tabla 4: Pruebas sobre distintos tamaños de población,

Sub-población (k)	Fitness	Tiempo [s]
50	51.688	507.024
100	48.978	434.575
150	50.173	591.193
200	50.608	639.207

Tabla 5: Pruebas sobre distintos tamaños de sub-población,

 2 El parámetro G representa la brecha generacional, h es el tamaño de la sub-población generada y N el tamaño de la población en cada generación.

 $^{^{3}}$ Si bien con N = 600 se obtuvo mayor adaptación, se recomienda utilizar N = 300 debido a que un tamaño superior incurre una reducción substancial en la performance.

Mutation Probability	Fitness	Height [m]
0.001	45.700	1.303
0.005	46.870	1.301
0.010	51.688	1.300
0.035	47.456	1.510
0.070	55.269	1.300
0.100	54.982	1.300

Tabla 6: Adaptación en función de la probabilidad de mutación, utilizando los parámetros más óptimos de las Tablas 1-5.