



INSTITUTO TECNOLÓGICO DE BUENOS AIRES

SISTEMAS DE INTELIGENCIA ARTIFICIAL

Trabajo Práctico Especial 3

Algoritmos Genéticos

16 de mayo de 2018

Índice general

Objetivos	2
Descripción del problema	2
Clases	2
Equipamiento	2
Altura	3
Ataque y Defensa	3
Implementación	3
Repositorio	4
Forma de entrega	4
Fecha de entrega	5
Grupos	5
Adicional	7

Objetivos

Se desea implementar un motor de algoritmos genéticos para obtener las mejores configuraciones de personajes de un juego de rol.

Descripción del problema

El juego consiste en personajes que tienen cierta clase, ciertas propiedades y cierto equipamiento. El objetivo es lograr la mejor configuración de ellas para optimizar el desempeño del personaje en el juego. Cualquier supuesto no descripto de este problema no será válido sin antes consultarlo con la cátedra (en un escenario real, sería “el cliente”).

Clases

En el juego actualmente existen 4 tipos de personajes: Guerreros, Arqueros, Defensores y Asesinos. Cada personaje debe lograr diferentes objetivos en cuanto a su desempeño en el *ataque* y la *defensa*.

1. **Guerrero:** Este personaje estará en el frente de batalla junto a los defensores. Al estar casi tan expuesto como ellos, deberá tener un desempeño equilibrado en cuanto al ataque.
 $\text{Desempeño} = 0.6 * \text{Ataque} + 0.4 * \text{Defensa}$
2. **Arquero:** Este personaje se ubicará en el final del batallón, atacando desde la distancia. Como tal, no precisa de técnicas de defensa, sino que su mayor prioridad es un ataque efectivo.
 $\text{Desempeño} = 0.9 * \text{Ataque} + 0.1 * \text{Defensa}$
3. **Defensor:** Este personaje deberá interponerse entre los ataques de sus enemigos y sus colegas. Su función primordial es defender a sus aliados.
 $\text{Desempeño} = 0.1 * \text{Ataque} + 0.9 * \text{Defensa}$
4. **Asesino:** Este personaje se infiltrará en las tropas enemigas, pasando desapercibido y sigilosamente terminar con sus vidas. Idealmente, no precisará de defenderse, pero deberá estar preparado para ello en el momento en que revelen su presencia para poder huir.
 $\text{Desempeño} = 0.7 * \text{Ataque} + 0.3 * \text{Defensa}$

Equipamiento

Las piezas de equipamiento (items) contienen ciertas características (stats): Fuerza, Agilidad, Pericia, Resistencia y Vida.

ATENCION! Cada grupo tendrá asignado una clase y un multiplicador de puntos de stats. El mismo se sorteará en clase y está definido en la sección Grupos.

Estos puntos definirán un coeficiente, redefiniendo las características del personaje en base a las características de sus items, de la siguiente manera:

$$\blacksquare \text{Fuerza}_p = 100 * \tanh(0,01 * \sum \text{Fuerza}_{item})$$

- $Agilidad_p = \tanh(0,01 * \sum Agilidad_{item})$
- $Pericia_p = 0,6 * \tanh(0,01 * \sum Pericia_{item})$
- $Resistencia_p = \tanh(0,01 * \sum Resistencia_{item})$
- $Vida_p = 100 * \tanh(0,01 * \sum Vida_{item})$

Se encuentran los archivos para cada set de items, en formato **tsv** (Tab-separated Values)

Altura

Aquí vemos modificadores de ataque (ATM) y de defensa (DEM). Estos se definirán en base a la altura,

- $ATM = 0,5 - (3h - 5)^4 + (3h - 5)^2 + h/2$
- $DEM = 2 + (3h - 5)^4 - (3h - 5)^2 - h/2$

Cada individuo inicial deberá poseer cierta altura, con una distribución uniforme [1.3m - 2.0m].

Ataque y Defensa

El ataque y la defensa quedará definido como:

- $Ataque = (Agilidad_p + Pericia_p) * Fuerza_p * ATM$
- $Defensa = (Resistencia_p + Pericia_p) * Vida_p * DEM$

Implementación

El motor de algoritmos genéticos deberá implementar:

- Operadores genéticos
 - Cruce
 - Cruce de un punto
 - Cruce de dos puntos
 - Cruce uniforme
 - Cruce anular
 - Mutación
 - **Uniforme**
Se mantiene la probabilidad de mutación en todas las generaciones
 - **No Uniforme**
La probabilidad de mutación cambia dependiendo de la generación
- Selección y reemplazo*

- Elite
- Ruleta
- Universal
- Boltzmann (al menos implementada con Ruleta)
- Torneos (ambas versiones)
- Ranking
- Métodos de reemplazo (todos los vistos en clase). Recordar los parámetros de dichos métodos.
- Criterios de corte
 - Máxima cantidad de generaciones
 - Estructura
 - Contenido
 - Entorno a un óptimo

(*) La selección deberá ser $A * (\text{método1}) + (1-A) * (\text{método2})$. El reemplazo deberá ser $B * (\text{método3}) + (1-B) * (\text{método4})$. De esta forma, A,B definen porcentajes [0.0-1.0] de selección entre diferentes métodos. De existir otros métodos híbridos, parametrizarlos de la misma forma.

Atención! Dado que son demasiados parámetros, se pide un archivo de configuración externo que lea TODOS los parámetros (no deberá cambiar el código para correr con diferentes configuraciones o diferentes personajes).

Aunque el análisis deberán hacerlo sobre el personaje asignado, se pide que pueda recibir los parámetros de cualquier otro personaje para correr el algoritmo.

Repositorio

Cada grupo cuenta con un repositorio para el desarrollo del trabajo, donde debe encontrarse todo el material entregado. La entrega también deberá realizarse a través del campus. La dirección de acceso es: <https://bitbucket.org/itba/sia-2018-N> donde N es su número de grupo [01-12] (mismo repositorio TPE1 y TPE2).

La entrega deberá poseer el tag **TPE3** (por favor, respeten el nombre estrictamente).

Forma de entrega

Toda la entrega será digital. Deberá contener:

- Informe (4 carillas A4 + anexo)
- Código fuente del trabajo
- Binario ejecutable
- El documento utilizado para realizar la presentación
- Un README.md o README.txt o simplemente README con una descripción del procedimiento necesario para compilar y ejecutar el programa
- La implementación deberá funcionar sobre el sistema operativo Linux
- Un archivo de configuración COMPLETO

Fecha de entrega

Miércoles 30 de Mayo a las 15:00hs.

Grupos

Aquí se definen los multiplicadores.

■ Guerrero 1

- $Fuerza_{item} * 1,1$
- $Agilidad_{item} * 0,9$
- $Pericia_{item} * 0,8$
- $Resistencia_{item} * 1,0$
- $Vida_{item} * 0,9$

■ Guerrero 2

- $Fuerza_{item} * 1,2$
- $Agilidad_{item} * 1,0$
- $Pericia_{item} * 0,8$
- $Resistencia_{item} * 0,8$
- $Vida_{item} * 0,8$

■ Guerrero 3

- $Fuerza_{item} * 0,8$
- $Agilidad_{item} * 0,9$
- $Pericia_{item} * 0,9$
- $Resistencia_{item} * 1,2$
- $Vida_{item} * 1,1$

■ Arquero 1

- $Fuerza_{item} * 0,8$
- $Agilidad_{item} * 1,1$
- $Pericia_{item} * 1,1$
- $Resistencia_{item} * 0,9$
- $Vida_{item} * 0,7$

■ Arquero 2

- $Fuerza_{item} * 0,9$
- $Agilidad_{item} * 1,1$
- $Pericia_{item} * 1,0$
- $Resistencia_{item} * 0,9$

- $Vida_{item} * 0,8$
- **Arquero 3**
 - $Fuerza_{item} * 0,8$
 - $Agilidad_{item} * 0,8$
 - $Pericia_{item} * 0,8$
 - $Resistencia_{item} * 1,1$
 - $Vida_{item} * 1,2$
- **Asesino 1**
 - $Fuerza_{item} * 0,8$
 - $Agilidad_{item} * 1,2$
 - $Pericia_{item} * 1,1$
 - $Resistencia_{item} * 1,0$
 - $Vida_{item} * 0,8$
- **Asesino 2**
 - $Fuerza_{item} * 0,9$
 - $Agilidad_{item} * 1,0$
 - $Pericia_{item} * 1,1$
 - $Resistencia_{item} * 1,0$
 - $Vida_{item} * 0,9$
- **Asesino 3**
 - $Fuerza_{item} * 0,9$
 - $Agilidad_{item} * 0,9$
 - $Pericia_{item} * 1,0$
 - $Resistencia_{item} * 1,1$
 - $Vida_{item} * 1,0$
- **Defensor 1**
 - $Fuerza_{item} * 1,0$
 - $Agilidad_{item} * 0,9$
 - $Pericia_{item} * 0,7$
 - $Resistencia_{item} * 1,2$
 - $Vida_{item} * 1,1$
- **Defensor 2**
 - $Fuerza_{item} * 1,1$
 - $Agilidad_{item} * 0,8$
 - $Pericia_{item} * 0,8$

- $Resistencia_{item} * 1,1$
- $Vida_{item} * 1,1$
- **Defensor 3**
 - $Fuerza_{item} * 0,9$
 - $Agilidad_{item} * 0,9$
 - $Pericia_{item} * 0,9$
 - $Resistencia_{item} * 1,0$
 - $Vida_{item} * 1,3$

Adicional

Adicionalmente (no forma parte de la evaluación), se pide una rutina que en N minutos logre una configuración (“raid”) de 10 personajes, maximizando su performance, mediante algoritmos genéticos, según las siguientes reglas:

- Cada clase diferente mejora la performance total del raid en un 5 %.
- Sólo hasta 4 personajes de la misma clase son permitidos.
- Sólo hasta 2 personajes con la misma configuración son permitidos.
- Deberán haber entre 1 y 2 defensores, para evitar el fracaso del objetivo (“wipe”).