Funcionamiento de la IA

1. Propósito

El propósito de este documento es explicar el diseño y la implementación de los jugadores controlados por la máquina que se han creado como respuesta a la historia de usuario (insertar identificador y nombre).

Para más información sobre implementaciones previas véase el apartado 6.

2. Terminología

- Nos referiremos a los jugadores automáticos como máquinas.
- Por estrategia nos referimos al atributo RunStrategy de la clase Máquina.
- Cuando aparece la palabra máquinas seguido de alguna de las palabras easy, medium o hard nos referimos a los jugadores automáticos que tienen como RunStrategy la identificada por ese nombre.
- Pivote: letra y coordenadas de una ficha en el tablero que se usará para concatenar una palabra

3. Funcionamiento general

A grandes rasgos, se ha decidido crear una clase Integrante de la que hereda la ya existente Jugador y una nueva clase Máquina. Esta nueva clase Máquina nos servirá para implementar estos jugadores automáticos.

Las máquinas tienen una estrategia. Esas estrategias tienen un método principal llamado run que devuelve una lista de comandos. Las máquinas utilizan sus estrategias para generar la lista de comandos que van a ejecutar en su turno.

Así, cuando en el orden de turnos le corresponde jugar a las máquinas estas simplemente tienen que generar la lista de comandos, ejecutarla y pasar su turno.

4. Estrategias

Para poder aplicar distintos niveles de dificultad era necesario crear un comportamiento diferente dependiendo del nivel deseado. Para ello hemos decidido utilizar el patrón estrategia, lo que nos permite tener una estrategia para las máquinas y simplemente hay qué decidir cuál de las disponibles se debe utilizar. De esta manera, si surge la necesidad de cambiar los niveles de dificultad o añadir otros comportamientos distintos gracias a este diseño sería muy sencillo.

La clase creada para esta implementación es la clase RunStrategy. Se ha decidido utilizar herencia para la implementación dado que los tres niveles actuales

utilizan tres métodos iguales y para evitar repetirlos en cada uno éstos se encuentran en esta clase principal. Esos tres métodos son:

- esVocal(char letra) que decide si el caracter letra es una vocal,
- generarComandos(Map<Coordenadas, Character> lado, Integrante maquina) que devuelve una lista de comandos ComandoColocarFicha para cada carácter en las coordenadas que le corresponden
- buscarPivotes(boolean vocal, ScoredWord palabra) que devuelve una lista pares de carácter con coordenadas. Esos pares representan una letra del tablero y sus coordenadas con las que el jugador automático intentará crear una palabra. Las letras que devuelve son las pertenecientes a la palabra que se le pasa que cumplen el criterio que se pide. Si el booleano vocal es true solo tomaremos las letras vocales, si es false tomaremos todas las letras de la palabra.

En este momento, las tres estrategias disponibles tienen un funcionamiento muy similar así que describiremos su funcionamiento general y se darán los detalles específicos para cada nivel en sus respectivos apartados.

El método run de las estrategias tiene como objetivo crear los comandos para colocar una palabra a partir de las letras que ha robado la máquina y pasar el turno.

Al comienzo del turno de una máquina existen dos posibles situaciones: que sea la primera palabra que se coloca en la partida o que no lo sea. La diferencia entre uno y otro es que en el primero no es necesario concatenar la palabra que vamos a colocar con las ya colocadas en el tablero dado que no hay ninguna colocada y al menos una ficha debe ocupar la casilla central.

Para esta primera opción, no es necesario concatenar así que obtenemos la lista de palabras que existen en el diccionario que se pueden formar con las fichas de la máquina y colocaremos una de forma horizontal partiendo de la casilla central en las coordenadas (7,7). La diferencia entre niveles de dificultad para esta parte se verá en sus respectivos apartados. Si no hay ninguna palabra que se pueda colocar el método devolverá una lista de comandos que sólo contendrá el ComandoPasarTurno.

Para la segunda opción la tarea se complica ya que tenemos que colocar una palabra concatenando con las ya puestas en el tablero. Para concatenar tendremos que usar una de las palabras existentes en el tablero así que primero obtenemos la lista de esas palabras. Mientras no encontremos una palabra que podamos colocar, realizaremos el procedimiento que se describe a continuación para cada una de las palabras del tablero. Para la palabra actual con la que queremos concatenar obtenemos los pivotes que acepta el nivel de dificultad. A continuación probaremos con cada pivote disponible. Ahora generamos una lista de palabras que existen en el diccionario con las letras de las fichas de la máquina y la letra del pivote (para más información véase el documento ImplementaciónDiccionario). Como último paso, comprobaremos si es posible

colocar la palabra en el tablero cumpliendo con las reglas del scrabble para cada una de estas palabras siguiendo el orden de prioridad del nivel de dificultad hasta encontrar alguna que nos sirva o nos quedemos sin palabras. En el caso de que encontremos una palabra que se puede colocar generamos una lista de comandos para colocar la palabra con el método generarComandos. En el caso de que no encontremos una palabra para ninguna de las posibilidades se devolverá una lista cuyo único comando es un ComandoPasarTurno.

Aclaración: Si se ve el código se ve que se llama dos veces al método generarComandos y que para ello se generan dos mapas distintos. Esto se debe a que una de las normas es que no se puede colocar una ficha si no tiene ninguna al lado. Para cumplir con esta regla, se genera un mapa ordenado de mayor a menor y otro de menor a mayor (utilizando las clases OrdenarCoordenadas y OrdenarCoordenadasInversa). De esta manera, al generar los comandos, uno devolverá una lista de comandos que va desde el pivote aumentando de uno en uno y otra que va desde el pivote disminuyendo de uno en uno.

4.1. EasyStrategy

Las máquinas easy tienen las siguientes características:

- Utilizan como pivotes únicamente las letras vocales
- Orden de preferencia de colocación: Este nivel de dificultad intentará colocar palabras preferentemente por su tamaño en este orden: tamaño 2, 3, 4, 5, 6, 7. De esta manera, en la mayoría de los casos las palabras no superarán la longitud de 3 letras salvo en casos extraños.

4.2. MediumStrategy

Las máquinas medium tienen las siguientes características:

- Utilizan como pivotes únicamente las letras vocales
- Orden de preferencia de colocación: Este nivel de dificultad intentará colocar palabras preferentemente por su tamaño en este orden: tamaño 4, 3, 2, 5, 6, 7. De esta manera, en la mayoría de los casos las palabras serán de longitud 4, en ocasiones de 3. Según avanza el juego es normal que empiezen a colocar de tamaño 2 al quedarse sin mucho espacio. Sólo en circunstancias extrañas se ven obligadas a colocar palarbas por encima de este tamaño.

4.3. HardStrategy

Las máquinas hard tienen las siguientes características:

- Utilizan como pivotes únicamente las letras vocales
- Orden de preferencia de colocación: Este nivel de dificultad intentará colocar palabras preferentemente por su tamaño en este orden: tamaño

6, 5, 4, 3, 2, 7. De esta manera, en la mayoría de los casos las palabras no superarán la longitud de 6 o 5 letras e irán reduciendo el tamaño si no encuentran nada. Es raro que se coloque una de 7. Al igual que para las máquinas médium, según avanza el juego es normal que se coloquen sobre todo palabras de longitud 2 al quedarse sin espacio.

5. Diagramas

Tenemos dos diagramas de secuencia, uno que muestra el desarrollo del turno de la máquina y otro que detalla más concretamente el funcionamiento del método run de las estrategias (véase en los diagramas UML IA_Secuencia_General e IA_Secuencia_Run).

También tenemos un diagrama de clases que muestra como se relacionan las clases para esta funcionalidad (véase en los diagramas UML IA_Clases).

6. Historial

Previamente hubo otra implementación. Esta implementación, en vez de colocar palabras en función de las fichas robadas por la máquina, no robaba fichas y elegía una palabra del diccionario al azar para colocarla, robaba las fichas que necesitaba para ello y la colocaba. Para ello simplemente se cogía una palabra al azar y se buscaba entre las letras del tablero para concatenar (si era la primera palabra se colocaba directamente), si esta no servía se seguía obteniendo palabras al azar hasta obtener alguna válida o hasta que se hubieran dado 100 intentos. Para diferenciar entre niveles de dificultad, las máquinas easy ponían una palabra, las máquinas médium tenían una probabilidad del 40% de colocar dos palabras y las hard tenían una probabilidad del 60% de colocar dos palabras y una probabilidad del 40% de comprar la ventaja saltar jugador en caso de tenerla.