

Práctica 5: Razonamiento Basado en Casos. Documento de Diseño

GRUPO: 01

ESTRUCTURA DE LOS CASOS

Enumerar los atributos de la descripción/consulta y el tipo de datos de cada uno.

Ms Pacman.

Posición de MsPacMan: Integer[1] (no se usa en la query solo para dividir la memoria) y su último movimiento: Move[1]

Distancia de pacman - fantasmas: Integer[4] y el movimiento relativo de pacman a los fantasmas: Move[4]. Se unen en 4 strings para poder compararlos juntos (distancia movimiento) ordenados por distancia (el primero es el del fantasma más cercano)

Tiempo que les queda de comestible a los fantasmas: Integer[4] y la distancia a pacman: Integer[4]. Se unen en 4 strings para poder compararlos juntos (distancia tiempo) ordenados por distancia (el primero es el del fantasma más cercano)

Distancia PPill más cercana: Double[1] y el movimiento relativo de pacman a la PPill: Move[1]. Se unen en un string para poder compararlo juntos (distancia movimiento).

Distancia a la pill más cercana: Double[1] y el movimiento relativo de pacman a la pill: Move[1]. Se unen en un string para poder compararlo juntos (distancia movimiento).

Nivel del mapa: Integer[1] (no se usa en la query solo para dividir la memoria)

Score: Integer[1] (Se usa solo en el result para decidir si guardar o no un caso)

Fantasmas.

Posición del Fantasma en cuestión: Integer[1] (no se usa en la query solo para dividir la memoria) y su último movimiento: Move[1]

Distancia de pacman - fantasmas (todos menos el que comparamos): Integer[3] y el movimiento relativo de pacman a los fantasmas: Move[3]. Se unen en 3 strings para poder compararlos juntos (distancia movimiento) ordenados por distancia (el primero es el del fantasma más cercano)

Tiempo que les queda de comestible a los fantasmas (todos menos el que comparamos): Integer[3] y la distancia a pacman: Integer[3]. Se unen en 3 strings para poder compararlos juntos (distancia tiempo) ordenados por distancia (el primero es el del fantasma más cercano)

Tiempo fantasma: Integer[1] Tiempo que la queda de comestible al fantasma del que queremos devolver el movimiento

Distancia pacman-fantasma + movimiento (distancia movimiento): Para el fantasma que estamos comparando.

Distancia P Pill más cercana a pacman: Double[1] y el movimiento relativo de pacman a la P Pill: Move[1]. Se unen en un string para poder compararlo juntos (distancia movimiento).

GhostType del fantasma del que queremos devolver el movimiento (Para calcular el score)

Nivel del mapa: Integer[1] (no se usa en la query solo para dividir la memoria)

Representación del resultado del caso (positivo, negativo, valor entre [-1,1], etc.

El resultado del caso será un número que indicará el progreso que se ha realizado.

El resultado puede ser negativo si el resultado ha sido especialmente malo

El resultado será un entero cuanto más alto mejor

En el caso de MsPacman

Afecta positivamente al resultado: Ganar puntos, pasar de nivel

Afecta negativamente al resultado: Perder una vida.

Fantasmas:

Acercarse a pacman / alejarse dependiendo de si es comestible o no

Matar a pacman

Que pacman no pase de nivel

PERSISTENCIA

Describir si se utilizará una única base de casos o algún tipo de partición por estado edible, oponente, nivel, etc (que puede coincidir o no con la indexación en memoria).

Usamos una base de casos genérica y una por cada oponente.

Las bases de casos estarán particionadas por nivel y dentro de cada nivel por junction aunque esto es solo en memoria, sólo existe un archivo por base de casos.

Indicar si la base de casos será genérica y/o especializada para los distintos contrincantes.

Una base de casos genérica que contendrá casos precalculados y una por cada oponente que se rellenará antes de la competición.

ORGANIZACIÓN EN MEMORIA

Describir si se utilizará una organización lineal en memoria o algún tipo de indexación o subdivisión de la base de casos.

La memoria estará subdividida por nivel del mapa y por junction (posición actual de pacman/el fantasma).

RECUPERACIÓN

Describir el proceso de recuperación según la organización en memoria. Indicar si para la recuperación se compararán todos los atributos de la consulta con los casos o solo los más relevantes.

Se accede a la partición correspondiente en base a los parámetros del caso actual, se recorren de forma lineal todos los casos de esta partición. Y nos quedamos con los casos más parecidos, en principio se comparan sólo algunos atributos de la descripción (aquellos que no se usan para indexar la memoria o para sacar el score).

Indicar si se utilizará pesos para comparar los atributos.

Sí. Los pesos se han asignado teniendo en cuenta la importancia relativa de cada uno.

Especificar las medidas de similitud, tanto local como global que se aplicará a cada atributo.

**Todas las comparaciones se modulan para encontrarse en el rango [0-1], las distancias se modulan respecto a una distancia máxima que será superior a cualquier distancia posible en el mapa*

Para las distancias y los tiempos se usan intervalos y se considera el movimiento relativo con la igualdad.

En el caso de la distancia de los fantasmas a pacman:

Si la distancia es grande el movimiento no se tiene en cuenta, si la distancia es mediana se hace el intervalos de distancias siempre que el movimiento relativo no sea igual al opuesto y si la distancia es pequeña el movimiento tiene que ser el mismo (si no devolvemos 0 en la similitud)

Para el tiempo de los fantasmas se hace un intervalo dependiendo de la distancia de pacman al fantasma. Si los fantasmas están lejos no nos importa su comestibilidad, si están a media distancia hacemos un intervalo y si están cerca: devolvemos similitud cero si uno es comestible y el otro no, si los dos son no comestibles devolvemos similitud 1 y si los dos son comestible un intervalo(200)

En la función de similitud global usamos la clase Average con los pesos definidos.

Indicar el número de vecinos más cercanos a recuperar. 5 vecinos más cercanos para la base de casos específica y 4 para la general

Indicar cuándo se recuperarán los casos (cambio de posición, nueva intersección, ...)

Los casos se recuperarán únicamente cuando una entidad (MsPacman o cualquier fantasma) llegue a una intersección, es decir, cuando podemos realizar un movimiento.

REUTILIZACIÓN

Indicar cómo se adaptará la solución o si se aplicará directamente. Especificar cómo se combinan las soluciones de los casos recuperados (votación, votación ponderada, ...) y si se tendrá en cuenta el resultado del caso para realizar dicha adaptación.

Votación ponderada, basándonos en la similitud del caso, la cantidad de votos que haya para un movimiento y el score que ha conseguido dicho movimiento. Nos quedamos con el que tenga mayor valor. Es decir, se da un puntuación a cada posible movimiento basada en los datos y similitud obtenidos después de aplicar el algoritmo KNN, el movimiento con mayor puntuación es el que se acaba devolviendo.

REVISIÓN

Indicar la función de cálculo del resultado del caso y cuando se calcularía (p.e. tras 5 intersecciones se calcula el cambio en el score).

MsPacMan

Se calcula al pasar 5 junctions.

Score = diferencia_ptos + cambio_nivel * 500 - vidas_perdidas * 200

Fantasmas.

Se calcula al pasar 10 junctions, conjunto todos los fantasmas (porque usamos un único CBR para todos cuando uno pasa por una junction cuenta)

Si el fantasma es comestible

Score = (Diferencia distancia * -20) + 300 * vidas_perdidas - 500 * CambioNivel

Si el fantasma no es comestible

Score = -(Diferencia distancia * +50) + 300 * vidas_perdidas - 500 * CambioNivel

Diferencia de distancia: la diferencia entre la distancia del pacman-fantasma en el tick actual y la distancia pacman-fantasma guardada en el caso. Es negativa si nos acercamos y positiva si nos alejamos.

RECUERDO

Indicar cual es la política de recuerdo de los casos (todos, solo los que no tengan vecinos muy similares, ...), y cuando se realiza dicho almacenamiento en ficheros (se recomienda que los nuevos casos se guarden al finalizar, aunque esto no es totalmente obligatorio, pero penaliza el tiempo de ciclo).

Cuando el caso más similar recuperado de la base de casos tenga una similitud inferior al 0,95 y siempre que el resultado del caso sea positivo para no llenar la base de casos con casos negativos (score > 120 para pacman y score > 200 para los fantasmas).

Consideramos que debemos guardar los casos en memoria desde el momento que sabemos que un caso es positivo y diferente al resto de casos de nuestra base de datos. La escritura en disco se realizará al finalizar la ejecución del programa para no penalizar el tiempo de ciclo, mientras tanto, los casos quedarán guardados en memoria.