

Proyecto 3 - Algoritmos genéticos

Yuen Law

Semestre I, 2020

Objetivos

1. Aplicar conceptos de algoritmos genéticos.
2. Aplicar conocimientos adquiridos en optimización de algoritmos.

Descripción

En este proyecto deberán usar algoritmos genéticos para crear robots que puedan atravesar un terreno de la mejor manera posible. Para esto, los robots contarán con un set de cromosomas que determinen su comportamiento y le ayuden a tomar decisiones conforme avance. Tendrá también otro set de cromosomas que determine el tipo de hardware con el que cuenta el robot.

Hardware

Cada robot cuenta con 3 elementos de hardware que se pueden configurar: la batería, la cámara y el motor. Cada uno de estos elementos tiene ventajas, desventajas y costo.

Batería El tamaño de la batería determinará la distancia máxima que el robot puede recorrer, la cual depende también del consumo de energía de los otros 2 componentes. Entre más grande, mayor la distancia, pero también mayor el costo. Existen 3 tamaños de batería.

Cámara La cámara determina la cantidad de espacios hacia adelante que el robot puede ver. Son 3 tipo de cámara. El modelo estándar le permite al robot ver un espacio en todas las direcciones y no consume energía adicional. El siguiente modelo le permite ver 2 espacios, y el último modelo le permite ver 3 espacios. Cada modelo mejorado consumirá más energía y tendrá un mayor costo.

Motor La potencia del motor determina el tipo de terreno (más información sobre los terrenos en la siguiente sección) por el cual el robot puede pasar sin problemas. Existen 3 tipo de motor, el modelo estándar solo puede atravesar por terreno normal y no consume energía adicional. El segundo modelo puede atravesar terrenos normales y moderados, pero consume más energía si pasa por estos últimos. El último modelo pasa por terreno normal, moderado y difícil y el

consumo de energía también es acorde al tipo de terreno. Además cada modelo mejorado tiene un costo adicional.

Comportamiento

El comportamiento del robot estará determinado por cadenas de Markov en donde los porcentajes son determinados por los cromosomas del robot. Es decir, todos los robots usarán el mismo modelo, lo que varía es el porcentaje de cada arco. El diseño del modelo queda completamente a su criterio, pero el modelo debe tener suficientes estados que le permitan al robot tomar decisiones basado en el terreno y el hardware que posee.

Terreno

El terreno consta de una matriz de 20×20 . El punto de inicio es la esquina inferior izquierda y el destino es la esquina superior derecha. Cada celda de la matriz tiene 1 de 4 tipos de terreno: normal, moderado, difícil y bloqueado. Cada robot puede pasar por los tipos de terreno según el motor que tenga equipado. Dependiendo de cómo se defina el comportamiento de los robots, es posible que un robot entre a un terreno para el cual no está equipado, en este caso, el recorrido del robot termina ahí. Las celdas bloqueadas son completamente inaccesibles, por lo ningún robot puede entrar. El consumo de energía se calcula al "salir" de la celda. Por ejemplo, pasar de una celda normal a una difícil no consumirá energía adicional. El mapa completo puede ser generado aleatoriamente, o leído de archivo, pero el mismo mapa será usado para todos los robots y todas las generaciones.

Requerimientos funcionales

El algoritmo genético debe tener todas las partes características de este tipo de algoritmos: población inicial, selección, cruce, mutación y nuevos individuos, además de la función de adaptabilidad, la cual debe considerar: distancia al objetivo, costo, y tiempo. Los parámetros, características y detalles de implementación quedan a criterio de ustedes, pero el algoritmo de producir individuos aceptables al final. Es decir, la última generación debe ser considerablemente mejor que la primera en el logro del objetivo.

Además deben incluir una interfaz gráfica que permita verificar cada una de las partes del algoritmo genético y la evolución de la población. Se espera al menos poder seleccionar un individuo en cualquier generación para examinarlo y ver: cómo fue producido (ver sus padres y los cromosomas correspondientes), su resultado de

adaptabilidad y su(s) recorrido(s) en el mapa. Debería ser entonces posible seleccionar un individuo en la última generación, ver su recorrido y rastrear su ascendencia hasta la primera generación. No es necesario que la verificación sea en tiempo real, es decir, pueden dejar correr el algoritmo hasta que termine, guardar la información necesaria para la verificación y presentar los resultados al final.

Entregables

1. Documentación PDF, en el formato IEEE para conferencias (máximo 8 páginas) con las siguientes secciones: Abstract, Introducción (planteamiento del problema), Trabajo relacionado, métodos (descripción de la solución), Resultados (análisis del tiempo de ejecución del algoritmo, comparado con la calidad de los resultados), Conclusión y Referencias.
2. Código en un repositorio de Git. El último commit realizado debe ser antes de la fecha y hora límite de entrega.

Aspectos administrativos

1. Fecha de entrega: lunes 10 de agosto, hora: 11:59 p.m.
2. El pdf debe ser enviado por medio de Schoology
3. El proyecto debe realizarse en parejas
4. El proyecto puede ser desarrollado en el lenguaje/herramienta de su elección.
5. Las tareas entregadas con retraso tendrán 10 puntos menos de la nota obtenida, por cada día de retraso. El primer día cuenta a partir de los primeros 15 minutos después de la hora especificada.

Evaluación

Rubro	Valor
Documento PDF	(30)
- Abstract	2.5
- Introducción	2.5
- Trabajo relacionado	2.5
- Métodos	10
- Resultados	7.5
- Conclusión	2.5
- Referencias	2.5
Código	(70)
- Diseño de cromosomas	10
- Diseño del comportamiento (Markov)	15
- Función de adaptabilidad y selección	10
- Cruce	5
- Mutación	5
- Mapa	5
- GUI	20
Total	100