Universidad Tecnológica Centroamericana
Facultad de Ingeniería

CC414 - Sistemas Inteligentes
Docente: Kenny Dávila, PhD

Tarea #2 (2% Puntos Oro)

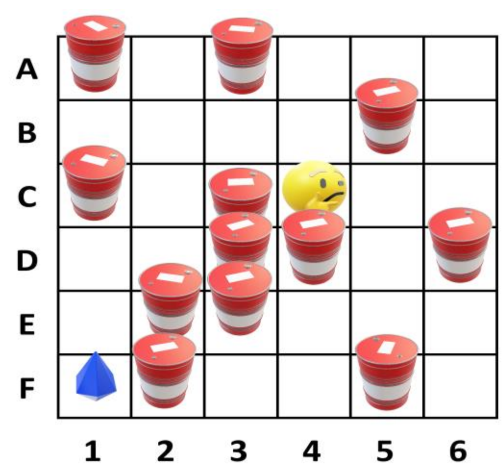
Parte 1. Buscando Soluciones (0.65 x 2 = 1.3)

Existen muchos problemas que se puede resolver mediante una secuencia de acciones. Es importante que un agente que actúa de manera racional intente identificar el curso de acción mas adecuado en cada situación en la que se encuentre. Hemos estudiado como la formulación correcta de un problema nos permitirá usar algoritmos de búsquedas que nos ayudan a identificar dichos cursos de acción de manera automática. Una formulación correcta de un problema consta de **5 componentes**: Estado inicial, Acciones por estado, Modelo de Transiciones, Prueba de Meta, y Costo de Camino. Adicionalmente, si tenemos una forma de estimar el costo de llegar de un estado x hacia un estado y (función heurística), entonces podemos aplicar búsquedas que prioricen las opciones mas prometedoras.

Para cada uno de los problemas siguientes, se le pide los siguiente:

1. **Generar una formulación completa del problema** incluyendo estado inicial, acciones, modelo de transiciones, prueba de meta y costo de camino.
2. **Proponer un heurístico válido** que ayude a estimar el costo de ir de un estado hacia otro.
3. **Mostrar los árboles de búsqueda generados** por las búsquedas **BFS** y **A***.

a) Un agente navegador que existe en una matriz de 6x6 cuadros y empieza en la ubicación mostrada en la figura, y debe llegar a la ubicación marcada con una pieza azul. No se permiten los movimientos en diagonal.



b) El juego de Sudoku tradicional requiere que los números del 1 al 9 se acomoden en una matriz de 9x9 donde ningún número se repite por columna o fila, lo que también implican que toda fila y columna deben tener exactamente una instancia de cada número. Adicionalmente, la matriz se subdivide en 9 grupos de 3x3 celdas, donde cada uno de estos grupos debe contener todos los números del 1 al 9 sin repetidos. Aquí se plantea una versión simplificada del juego usando los números del 1 al 4 y una matriz de tan solo 4x4, con 4 grupos de 2x2. Usando búsquedas, el agente debe completar el siguiente juego:

3	4	1	2
1			4
4			1
	1	4	

Parte 2. Descenso de Gradiente y Recocido Simulado (0.70)

En muchos problemas el objetivo es encontrar una configuración de parámetros o variables que minimizan o maximizan una función objetivo. Este proceso se conoce como optimización. Una técnica comúnmente utilizada para encontrar el mínimo/máximo de una función es el descenso de gradiente. Asumiendo que el valor óptimo de la función se encuentra dentro de un rango conocido, esta técnica empieza en una posición aleatoria dentro de dicho rango y utiliza el gradiente (derivadas parciales) de la función objetivo para estimar la dirección donde el gradiente cambia más rápido (e.g. es más bajo elevado). Si se desea minimizar la función, entonces uno puede moverse en dirección opuesta a la dirección donde el gradiente es mas elevado. Este movimiento se basa en un valor α que determina la longitud del “paso” que se da en la dirección indicada. Pasos largos nos ayudan a aproximarlos el valor optimo mas rápido, pero si el valor esta muy cerca es posible terminar dando vueltas sin llegar a la meta. Por otro lado, usar pasos cortos nos ayuda a acercarnos aún más a la meta, pero se requiere una mayor cantidad mayor de pasos. Por esta razón, el recocido simulado (simulated annealing) se basa en iniciar con pasos largos y gradualmente utiliza pasos más y más pequeños.

En este ejercicio usted deberá intentar encontrar el valor mínimo de la siguiente función:

$$f(x, y) = (x + 2y - 7)^2 + (2x + y - 5)^2$$

Para esto, deberá utilizar 5 intentos aleatorios (Random Re-starts) de descenso de gradiente con recocido simulado. Para cada intento, inicialice las variables x y y usando valores aleatorios entre -10 y 10. Luego, calcule el valor del gradiente en ese punto utilizando las derivadas parciales de $f(x, y)$:

$$\frac{d}{dx}f(x,y) = 10x + 8y - 34$$

$$\frac{d}{dy}f(x,y) = 8x + 10y - 38$$

Puede actualizar la posición del punto en la búsqueda local usando la regla de descenso de gradiente para minimización:

$$x \leftarrow x - \alpha\left(\frac{d}{dx}f(x,y)\right)$$

$$y \leftarrow y - \alpha\left(\frac{d}{dy}f(x,y)\right)$$

Donde el parámetro α será controlado mediante recocido simulado. Esto es, durante las primeras 5 iteraciones, el valor de α será 0.2, y luego se reducirá a 0.1 por 10 iteraciones más, luego será 0.05 por 10 iteraciones y finalmente será 0.01 durante las 10 iteraciones finales.

Puede usar 5 tablas para ilustrar el proceso, una tabla por cada reinicio aleatorio. Por cada fila puede indicar el valor inicial de x y y, el valor actual de α , los valores del gradiente dx y dy, y finalmente los valores actualizados de x, y al terminar la iteración.

Otras políticas

1. Esta tarea deberá trabajarse y entregarse **individualmente.**
2. La entrega será un solo documento en formato PDF. Dicho documento puede contener imágenes (fotos o scans) para las respuestas de aquellos ejercicios que resulte más fácil completar a mano.
3. El **plagio** será penalizado de manera severa.
4. Los estudiantes que entreguen una tarea 100% original recibirán una nota parcial a pesar de errores existentes. En cambio, los estudiantes que presenten tareas que contenga material plagiado recibirán 0% automáticamente independientemente de la calidad.
5. Tareas entregadas después de la fecha indicada solamente podrán recibir la mitad de la calificación final. Por esta razón, es posible que **un trabajo incompleto pero entregado a tiempo termine recibiendo mejor calificación que uno completo entregado un minuto tarde**.