Universidad del Bío-Bio
Facultad de Ciencias Empresariales
Departamento de Sistemas de Información

Problema 1

Suponga que se dispone de un agente que se mueve en una cuadrícula bidimensional compuesta de celdas cuadradas, las cuales pueden o no estar llenas de piedras. El agente se puede mover al norte, sur, este u oeste, una posición. Una de las celdas vacías contiene una caja la cual puede ser movida a una celda adyacente si el agente se para en la celda vecina a la misma, y luego se mueve en la dirección de la caja.

La caja no puede moverse de ninguna otra manera, lo cual implica que si la caja se mueve hacia una esquina, la misma no podrá ser retirada de la misma. Una de las celdas vacías se marca como la celda objetivo a la cual se desea llevar la caja.

P	P	P	P	P	P	P	P	P	P	P
P	X	P	P							P
P		P		P			P	P	P	P
P					C					P
P		P	P	P	P	P	P			P
P						A				P
P	P	P	P	P	P	P	P	P	P	P

Considere que el agente necesita el doble de energía al empujar la caja que cuando realiza otro tipo de acción.

P: hay piedras en esa celda
C: celda donde se encuentra la caja
A: posición inicial del agente
X: celda objetivo

Se pide plantear formalmente este problema

Problema 2

Se tiene tres monedas sobre una mesa en posición cara-sello-cara. Considere una manera de jugar que consiste en dar vuelta un par de monedas cada vez. Se desea que las tres monedas queden en la configuración sello-sello-sello en exactamente tres jugadas. Se pide plantear formalmente el problema

Problema 3

Se tiene un puzzle formado por tres fichas negras, tres fichas blancas y un espacio en blanco, colocados inicialmente como en la siguiente figura:

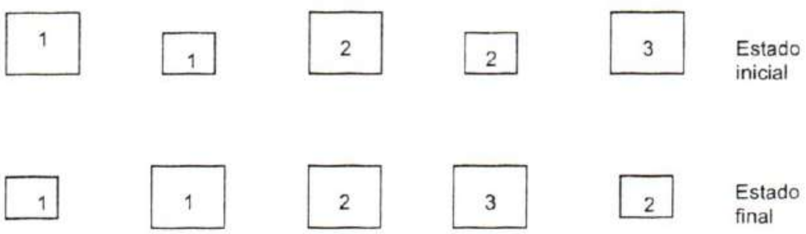
N N N   B B B

En este juego una ficha puede moverse a una posición adyacente vacía. En este caso, el costo es 1. Además una ficha puede saltar sobre una o dos fichas hasta alcanzar el espacio vacío. En este caso el costo es igual al número de fichas saltadas. El objetivo consiste en conseguir que todas las fichas blancas estén a la izquierda de las negras. La posición del espacio en blanco no tiene importancia.

Para este problema se pide plantearlo formalmente

Problema 4

Este problema consiste en modificar la posición inicial de un grupo de cajas hasta llegar a otra posición previamente fijada. El grupo de cajas está formado por tres cajas grandes y dos pequeñas, todas enumeradas y situadas formando una fila. Los estados inicial y final son los indicados en las siguientes figuras:



Se pide planearlo formalmente

1) Se representará las posiciones de la región mediante pares (fila, columna) siendo (1,1) la esquina superior izquierda y (7,11) la esquina inferior derecha.

Estados:

Los representaremos estados como una terna:

(posagente, poscaja, pospiedras) donde

posagente = (filaA, colA) representa la posición del agente

poscaja = (filaC, colC) representa la posición de la caja.

pospiedras = a un conjunto de posiciones (fila, columna) de las piedras

Estado Inicial:

((6,7), (4,6), {(1,1), (1,2), (1,3), (1,4), (1,5), (1,6), (1,7), (1,8), (1,9), (1,10), (1,11), (2,1), (2,3), (2,4), (2,11), (3,1), (3,3), (3,5), (3,8), (3,9), (3,10), (3,11), ..., (7,11)})

Estado Final:

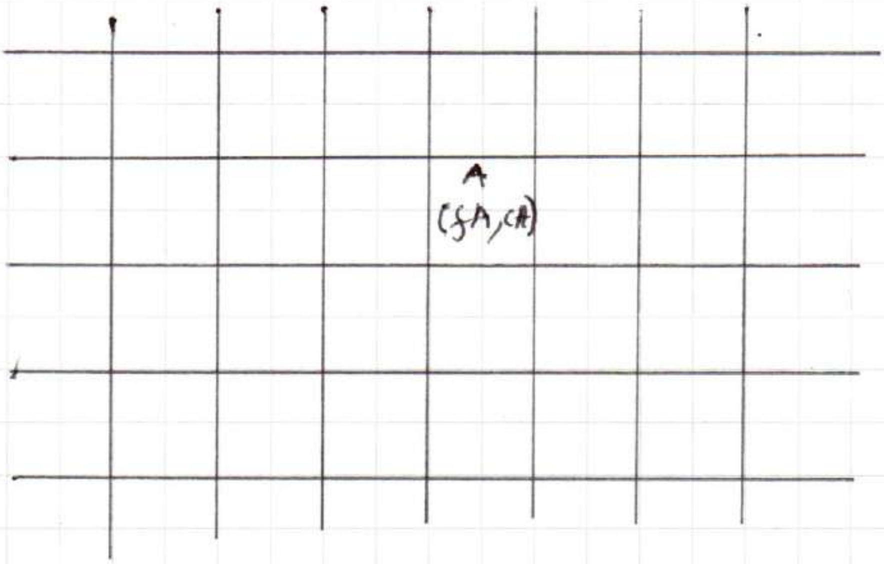
(—, (2,2), { % })
— = cualquier estado

Operadores:

Como los bordes de la cuadrícula están cerrados por piedras, no es necesario verificar los límites de la misma en los operadores.

Debemos distinguir en términos generales dos operadores cuyos efectos es mover el agente y mover la caja:

- Movimientos del Agente:



O1) Mover izquierda:

Si $(filaA, colA-1) \notin pospiedras \;\wedge$

$(filaA, colA-1) \neq poscaja \Rightarrow$

$((filaA, colA-1), poscaja, pospiedra)$

02) Mover Derecha

Si $(filaA, colA+1) \notin Pospiedras \wedge$
$(filaA, colA+1) \neq poscaja \Rightarrow$

$((filaA, colA+1), poscaja, pospiedras)$

03) Mover Arriba

Si $(filaA-1, colA) \notin Pospiedras \wedge$

$(filaA-1, colA) \neq poscaja \Rightarrow$

$((filaA-1, colA), poscaja, pospiedras)$

04) Mover Abajo

Si $(filaA+1, colA) \notin Pospiedras \wedge$

$(filaA+1, colA) \neq poscaja \Rightarrow$

$((filaA+1, colA), poscaja, pospiedras)$

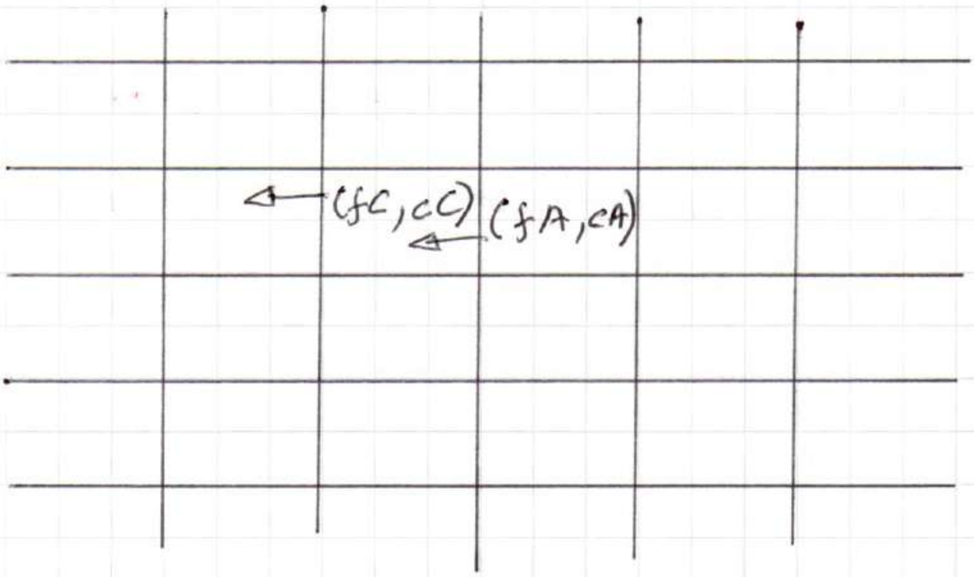
- Movimientos de la caja:

05) Empujar Izquierda:

Si (fila A, col A − 1) = poscaja ∧

(fila C, col C − 1) ∉ pospiedras ⟹

((fila A, col A − 1), (fila C, fila C − 1), pospiedras)



06) Empujar Derecha

Si (fila A, col A + 1) = poscaja ∧

(fila C, col C + 1) ∉ pospiedras ⟹

((fila A, col A + 1), (fila C, col C + 1), pospiedra)

07) Empujar Arriba

Si (fila A − 1, col A) = poscaja ∧
(fila C − 1, col C) ∉ pospiedras ⟹

((fila A − 1, col A), (fila C − 1, col C), pospiedras)

08) Empujar Abajo

Si (filaA+1, colA) = poscaja ∧

(filaC+1, colC) ∉ pospiedra ⟹

((filaA+1, colA), (filaC+1, colC), pospiedras)

Costo de Ruta:

Considerando la energía adicional que el agente necesita para empujar la caja, se proponen los siguientes costos de aplicación de los operadores:

01 a 04 : costo de 1

05 a 08 : costo de 2

$$c(op) = \begin{cases} 1 \text{ si } op = 01\text{-}04 \\ 2 \text{ si } op = 05\text{-}08 \end{cases}$$

Prueba de meta:

Comparar el estado actual con el estado final, es decir con

(−, (2,2), pospiedras)

o bien si

poscaja = (2,2)

Problema de las fichas blancas y negras.

Los estados los podemos definir por una 7-upla, una componente por cada ficha, incluyendo el espacio ~~en blanco~~ vacío. Las componentes tomarán los valores: N si la ficha es negra, V ~~si está~~ para representar el espacio ~~en~~ vacío, y B para indicar las fichas blancas.

$EE = \{ (c_1, c_2, c_3, c_4, c_5, c_6, c_7) \,/\, c_i \in \{N, V, B\},\ i = \overline{1,7} \}$

$EI : (N, N, N, V, B, B, B)$

$EF : (B, B, B, N, N, N, V)$

Operadores:

O1: mover_adyacente(i, j) : mueve una ficha i a una posición j

Precondiciones:

i) Si $|i - j| = 1$, es decir las ~~ficha~~ posiciones i y j son adyacentes

ii) $c_j = V$, es decir ~~en~~ la posición j está vacía.

iii) $1 \leq i \leq 7$ y $1 \leq j \leq 7$

Post-condiciones:

i) $c_j = c_i$, ~~es decir, en la posición j ponemos el espacio vacío que estaba~~ , es decir, en la posición j ~~colocamos la~~ queda la ficha que estaba en la posición i

ii $c_i = V$, es decir la posición i queda vacía.

OP2:

mover_salto1(i,j) : mueve una ficha desde la posición i a la posición j(V) saltando otra ficha intermedia

Precondiciones:

i) $|i-j|=2$, es decir, la ficha de la posición i está separada por una ficha del espacio ~~en blanco~~ vacío j

ii) $c_j = V$, es decir la posición j está ~~en~~ vacía.

iii) $1 \le i \le 7$, $1 \le j \le 7$

Postcondiciones:

i) $c_j = c_i$, es decir, en la posición j queda la ficha de la posición i

ii) $c_i = V$ en la posición i queda el espacio vacío.

OP3:

mover_salto2(i,j) : mueve una ficha desde la posición i a la posición j (V) saltando dos fichas intermedias.