

Inteligencia Artificial

Certamen # 1

Profesor: María Cristina Riff
Ayudante: Francisco Saravia Stein

20 de abril de 2007

1. Papers: 10 puntos

1. ¿Qué se entiende por restricciones duras o “hard constraints”?

Respuesta:

Son las restricciones que deben ser satisfechas, de lo contrario no estamos frente a una solución factible. Se contraponen a las restricciones blandas, que al no ser satisfechas se obtiene una solución de peor calidad, pero igualmente factible. (3 puntos)

2. ¿Por qué en AC-3 luego de aplicar $REVISE(V_k, V_m)$ no es necesario agregar el arco (V_m, V_k) ?

Respuesta:

Porque (V_m, V_k) ya se revisó o bien está en la cola Q y será revisado más adelante, entonces agregarlo no constituye un aporte. (4 puntos)

3. ¿Qué es un agente autónomo?

Respuesta:

Es un sistema situado en un entorno, que forma parte (siente mediante sensores) y actúa (mediante efectores) sobre su entorno en pos de su propio objetivo (o agenda). (3 puntos)

2. Materia: 40 puntos

1. Comente si es verdadero o falso justificando:

- a) Si un problema no tiene solución nunca podrá ser k -consistente.

Respuesta:

Verdadero. Sólo en el caso que $k < n$, donde n es el número de variables del problema.

- b) La heurística de la variable más conectada y con dominio más pequeño dinámico ayuda a que GBJ mejore su búsqueda.

Respuesta:

Falso. La heurística de variable más conectada sí ayuda, pero el dominio más pequeño dinámico no, porque no hay cambios en el dominio de las variables que produzcan re-ordenamiento.

- c) Se pueden mezclar técnicas de look-ahead para hacer más eficiente la búsqueda de soluciones.

Respuesta:

Falso. Las técnicas de look-ahead se pueden mezclar con técnicas de back-jumping.

- d) La explosión combinatorial está relacionada con el tamaño del grafo de restricciones.

Respuesta:

Falso. Está relacionada con el tamaño del dominio de las variables.

- e) El árbol de búsqueda depende de la técnica que se usa para resolver un CSP más que del modelo asociado.

Respuesta:

Verdadero. Cada técnica busca que el árbol de búsqueda sea menos frondoso, el modelo es el que define el espacio de búsqueda.

2. ¿Cuál es el objetivo de modelar un problema de optimización combinatorial?

Respuesta:

Es entender el problema y tener un mecanismo de comunicación. Además se pueden identificar semejanzas con problemas clásicos para usar las mismas técnicas de resolución.

3. Compare dos formas de definir variables y dominios para resolver un **sudoku** de tamaño 9. Determine el espacio de búsqueda de cada uno. ¿Cuál le parece más apropiado?

Respuesta:

Modelo 1:

X_{ij} = el número que está en la casilla i, j

$i \in [1, 9]$

$j \in [1, 9]$

$X_{ij} \in [1, 9]$

Tamaño e.b. = $9^{9 \cdot 9}$

Modelo 2:

$X_{ij} = \begin{cases} 1 & \text{en la casilla } i \text{ está el nro. } j \\ 0 & \text{si no} \end{cases}$

$i \in [1, 81]$

$j \in [1, 9]$

$X_{ij} \in \{0, 1\}$

Tamaño e.b. = 2^{9^3}

Como $2^{9^3} \gg 9^{81}$, la mejor elección, en términos de menor espacio de búsqueda, es el primer modelo.

3. Modelo: Problema de Cenar (25 puntos)

Varias familias salen a cenar juntas. Para aumentar la interacción social, desean sentarse en mesas de tal forma que no estén dos miembros de la misma familia en la misma mesa. Suponga que hay q mesas disponibles y que la j -ésima mesa tiene una capacidad de $b(j)$ personas. Suponga además que son p familias y que la i -ésima familia la integran $a(i)$ personas.

1. Modele el problema

Respuesta:

Se tienen p familias con $a(i)$ integrantes cada una.

Se tienen q mesas con una capacidad de $b(j)$ personas cada una.

Supuesto: $a(i) \geq q \quad \forall i$: La cantidad de mesas disponibles es mayor o igual a la cantidad de integrantes de la familia más numerosa.

Variable:

$$X_{ij} = \begin{cases} 1 & \text{persona de la familia } i \text{ está sentada en la mesa } j \\ 0 & \text{si no} \end{cases}$$

Restricciones:

$$\sum_i^p X_{ij} \leq b(j) \quad \forall j \in [1, q] \quad \text{Capacidad de la mesa } j$$

$$\sum_j^q X_{ij} = a(i) \quad \forall i \in [1, p] \quad \text{todas las personas deben estar sentadas}$$

$$\text{supuesto: } \sum_j^q b(j) \geq \sum_i^p a(i)$$

$$\sum_i^p X_{ij} \leq 1 \quad \forall j \in [1, q] \quad \text{A lo más una persona de la familia } i \text{ en cada mesa } j$$

4. Resolución: 25 puntos

Sea el siguiente CSP:

$$X_1 \leq X_2 \quad (1)$$

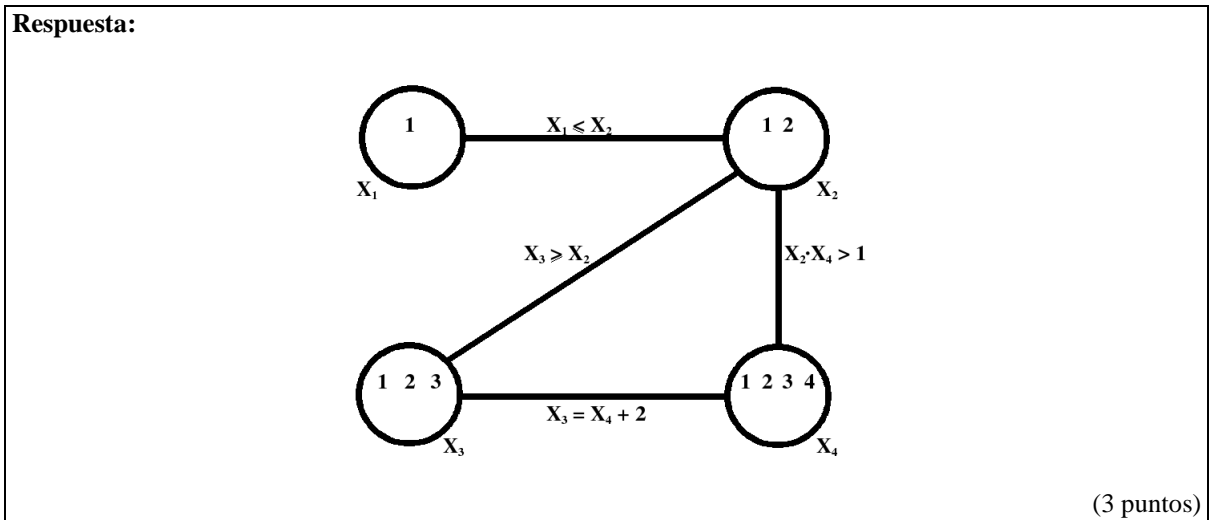
$$X_2 \leq X_3 \quad (2)$$

$$X_3 = X_4 + 2 \quad (3)$$

$$X_4 \cdot X_2 > 1 \quad (4)$$

Con dominios $D_1 = \{1\}$, $D_2 = \{1, 2\}$, $D_3 = \{1, 2, 3\}$, $D_4 = \{1, 2, 3, 4\}$

1. Dibuje el grafo de restricciones



2. Determine cuál de los siguientes métodos es más eficiente para encontrar **dos** soluciones al problema: CBJ, GBJ. Use el siguiente orden: X_1, X_3, X_2, X_4 y los dominios de izquierda a derecha. Para ello use una tabla con la siguiente estructura:

Variable	Instanciación	Filtro (Vars-dominios resultantes)	Pto. Backtrack

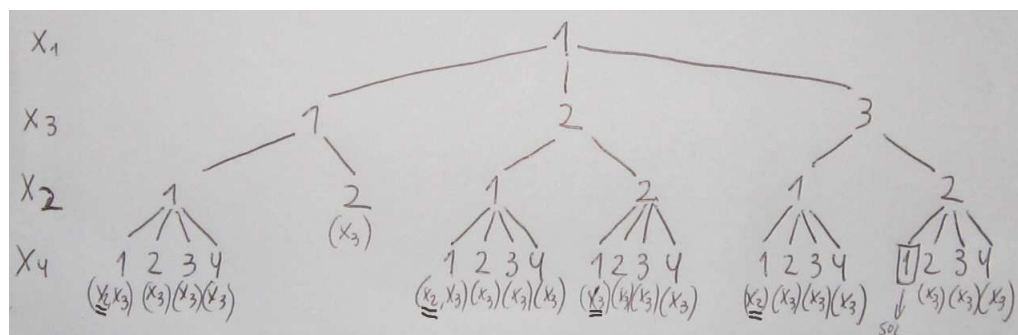
Dibuje el árbol de búsqueda para ambos métodos, analizando el trashing de cada uno.

Respuesta:
CBJ

Variable	Instanciación	Filtro	Conflictos	Pto. Backtrack
X_1	1			
X_3	1			
X_2	1			
X_4	1		$\{X_2, X_3\}$	
X_4	2		$\{X_3\}$	
X_4	3		$\{X_3\}$	
X_4	4		$\{X_3\}$	X_2
X_2	2		$\{X_3\}$	X_3
X_3	2			
X_2	1			
X_4	1		$\{X_2, X_3\}$	
X_4	2		$\{X_3\}$	
X_4	3		$\{X_3\}$	
X_4	4		$\{X_3\}$	X_2
X_2	2			
X_4	1		$\{X_3\}$	
X_4	2		$\{X_3\}$	
X_4	3		$\{X_3\}$	
X_4	4		$\{X_3\}$	X_3
X_3	3			
X_2	1			
X_4	1		$\{X_2\}$	
X_4	2		$\{X_3\}$	
X_4	3		$\{X_3\}$	
X_4	4		$\{X_3\}$	X_2
X_2	2			
X_4	1		Solución	
X_4	2		$\{X_3\}$	
X_4	3		$\{X_3\}$	
X_4	4		$\{X_3\}$	Fin del esp. búsqu.

Existe sólo una solución al problema, sin embargo, como se pide buscar 2, el algoritmo debe recorrer todo el espacio de búsqueda, hasta darse cuenta que no existen dos soluciones.

La única solución es $(X_1, X_2, X_3, X_4) = (1, 2, 3, 1)$



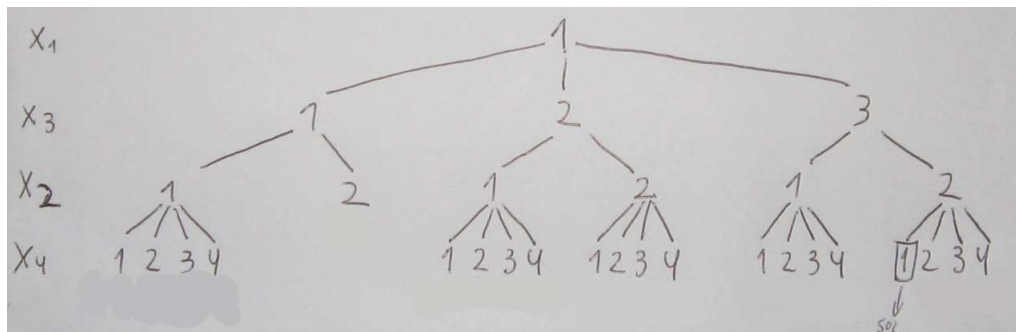
(4 puntos)

Respuesta:
GBJ

Variable	Instanciación	Filtro	Pto. Backtrack
X_1	1		
X_3	1		
X_2	1		
X_4	1		
X_4	2		
X_4	3		
X_4	4		X_2
X_2	2		X_3
X_3	2		
X_2	1		
X_4	1		
X_4	2		
X_4	3		
X_4	4		X_2
X_2	2		
X_4	1		
X_4	2		
X_4	3		
X_4	4		X_2
X_2	—		X_3
X_3	3		
X_2	1		
X_4	1		
X_4	2		
X_4	3		
X_4	4		X_2
X_2	2		
X_4	1		Solución
X_4	2		
X_4	3		
X_4	4		Fin del esp. búsq.

Existe sólo una solución al problema, sin embargo, como se pide buscar 2, el algoritmo también debe recorrer todo el espacio de búsqueda, hasta darse cuenta que no existen dos soluciones.

La única solución es $(X_1, X_2, X_3, X_4) = (1, 2, 3, 1)$



(4 puntos)

Respuesta:

Conclusiones:

Para dos soluciones hay que recorrer todo el árbol.

Los retornos son similares a los de BT, dada la configuración del grafo.

La cantidad de chequeos es similar.

GBJ no se preocupa de manejar una lista de conflictos, por lo que podría ser mejor.

CBJ hizo en una oportunidad un retorno a una variable de más arriba (de X_4 a X_3), también puede ser considerado mejor por esta razón.

El trashing de cada técnica es el mismo.

Ambos métodos generan árboles similares para este problema.

(1 punto)

3. Encuentre **una** solución del problema usando MFC (*Minimal Forward Checking*, donde la idea es sólo encontrar un valor consistente en cada dominio conectado futuro, posponiendo los otros chequeos de restricciones hasta que sea necesario). Use el orden X_1, X_2, X_3, X_4 y los dominios de izquierda a derecha.

Respuesta:

Minimal Forward Checking (MFC)			
Variable	Instanciación	Filtro	Retorno
x_1	1	$x_2 = \{1, 2\}$	
x_2	1	$x_3 = \{1, 2, 3\}$ $x_4 = \{\cancel{X}, 2, 3, 4\}$	
x_3	1	$x_4 = \{\cancel{2}, \cancel{X}, \cancel{X}\}$	
x_3	2	$x_4 = \{\cancel{2}, \cancel{X}, \cancel{X}\}$	
x_3	3	$x_4 = \{\cancel{2}, \cancel{X}, \cancel{X}\}$	x_2
x_2	2	$x_3 = \{\cancel{X}, 2, 3\}$ $x_4 = \{1, 2, 3, 4\}$	
x_3	2	$x_4 = \{\cancel{X}, \cancel{2}, \cancel{X}, \cancel{X}\}$	
x_3	3	$x_4 = \{1, 2, 3, 4\}$	
x_4	1	Solución encontrada: $(X_1, X_2, X_3, X_4) = (1, 2, 3, 1)$	

Como se pide sólo una solución, el algoritmo termina al encontrarla.

(5 puntos)

4. Determine la red arco-consistente usando AC-3. Muestre los elementos del dominio reducidos en cada paso y el arco responsable de su eliminación usando el siguiente orden de cola: de menor a mayor tasa de satisfacción de la restricción, donde la tasa de satisfacción de una restricción se calcula como:

$$\frac{\text{Número de instancias que permiten la satisfacción de la restricción}}{\text{Número total de instancias posibles de la restricción}}$$

Respuesta:

Ordenamiento de los arcos		
Arco	Instancias	Tasa
$X_1 - X_2$	(1, 1)(1, 2)	2/2
$X_2 - X_3$	(1, 1)(1, 2)(1, 3) (2, 1) (2, 2)(2, 3)	5/6
$X_2 - X_4$	(1, 1) (1, 2)(1, 3)(1, 4)(2, 1)(2, 2)(2, 3)(2, 4)	7/8
$X_3 - X_4$	(1, 1) (1, 2) (1, 3) (1, 4) (2, 1) (2, 2) (2, 3) (2, 4) (3, 1) (3, 2) (3, 3) (3, 4)	1/12

(3 puntos)

Arco consistencia:

La nodo consistencia no tiene efecto, ya que no hay restricciones unarias.

$$Q = \{(3 - 4), (4 - 3), (2 - 3), (3 - 2), (2 - 4), (4 - 2), (1 - 2), (2 - 1)\}$$

AC-3		
Revise	Filtro	Agregar a Q
(3 - 4)	$X_3 = \{\cancel{1}, \cancel{2}, 3\}$	(2 - 3)
(4 - 3)	$X_4 = \{1, \cancel{2}, \cancel{3}, \cancel{4}\}$	(2 - 4)
(2 - 3)	$X_2 = \{1, 2\}$	
(3 - 2)	$X_3 = \{3\}$	
(2 - 4)	$X_2 = \{\cancel{1}, 2\}$	(1 - 2) , (3 - 2)
(4 - 2)	$X_4 = \{1\}$	
(1 - 2)	$X_1 = \{1\}$	
(2 - 1)	$X_2 = \{2\}$	
(3 - 2)	$X_3 = \{3\}$	

Dado que todos los dominios quedaron de tamaño 1, el filtrado que realiza AC-3 encontró, además, la única solución al problema.

(5 puntos)