USTHB - FEI - Département d'Informatique
Master 2 ''Systèmes Informatiques Intelligents'' 2017/2018
Module ''Programmation par Contraintes''
Date de l'interrogation : 17/12/2017 <9h40-11h10>

Corrigé de l'interrogation

Exercice 1 (8 points=1+3+2+2)

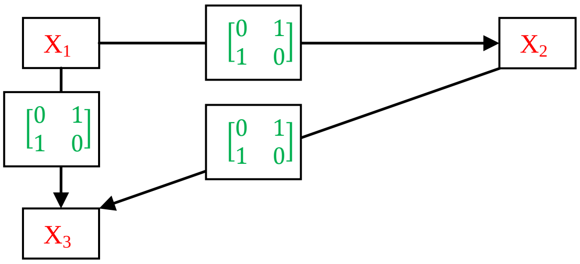
Le CSP binaire discret P=(X,D,C) suivant modélise le problème de coloriage d'un triangle avec deux couleurs :

- $X=\{X_1,X_2,X_3\}$
- $D(X_1)=D(X_2)=D(X_3)=\{1,2\}$
- $C=\{c_1 : X_1 \neq X_2,\ c_2 : X_1 \neq X_3,\ c_3 : X_2 \neq X_3\}$

1) Donnez la représentation graphique de P.
2) Le CSP est trivialement inconsistant. Montrez cette inconsistance avec l'algorithme de recherche Look-Ahead.
3) Donnez la représentation matricielle M_P de P.
4) Le CSP étant de taille inférieure ou égale à trois (ici trois), un algorithme de consistance de chemin tel que PC2 peut en détecter l'inconsistance. Montrez comment la détection se fait.

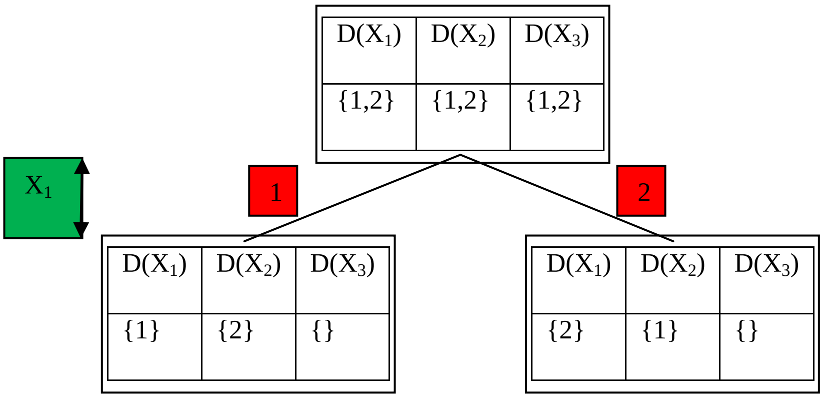
Solution :

1) Représentation graphique de P :



2) **Look Ahead :**

Le CSP est trivialement inconsistant : on ne peut pas colorier les sommets d'un triangle avec seulement deux couleurs sans que deux des trois sommets aient la même couleur.



Look Ahead détectera l'inconsistance de P de la façon suivante :

a. La racine de l'arbre de recherche sera étiquetée par le résultat du prétraitement du CSP de départ par AC3. Le prétraitement ne modifiera aucunement le CSP qui est déjà consistant d'arc (très visible sur la représentation graphique : le poids de chaque arc, qui est une matrice booléenne 2x2, a des 1 sur chaque ligne et des 1 sur chaque colonne). Le raffinement étiquetant un nœud est donné par l'évolution des domaines des variables. Après le prétraitement au niveau de la racine, les trois variables restent donc disjonctives avec le domaine commun {1,2}. On choisit une de ces trois variables comme première variable à instancier, et on opte pour X_1.

b. On instancie X_1 avec la première valeur du domaine, la valeur 1 (fils gauche de la racine) : les domaines du raffinement étiquetant le fils gauche de la racine, avant filtrage par AC3, vont ainsi être comme suit :

X_i	X_1	X_2	X_3
$D(X_i)$	{1}	{1,2}	{1,2}

c. On applique le filtrage par AC3, avec la file initialisée comme suit : $Q=\{(X_2,X_1),(X_3,X_1)\}$. On passe de la racine, dont le raffinement l'étquetant est consistant d'arc, au fils gauche en « perturbant » une seule variable, ici la variable X_1 : la file du filtrage par AC3 doit donc être initialisée aux arcs (X_k,X_1) sur lesquels il y a une contrainte (la valeur 2 qui ne figure plus dans le domaine de X_1 pourrait être, dans le raffinement de la racine, l'unique support de valeurs de certaines variables liées à X_1 par une contrainte, auquel cas le filtrage au niveau du fils gauche supprimerait ces valeurs des domaines de ces variables).

d. On prend l'arc (X_2,X_1) pour propagation : Q devient $Q=\{(X_3,X_1)\}$

e. La valeur 1 de $D(X_2)$ n'ayant pas de support dans $D(X_1)$, on la supprime de $D(X_2)$ et on rajoute la paire (X_3,X_2) à la file : $Q=\{(X_3,X_1),(X_3,X_2)\}$ et l'évolution des domaines est comme suit :

X_i	X_1	X_2	X_3
$D(X_i)$	{1}	{2}	{1,2}

f. On prend l'arc (X_3,X_1) pour propagation : Q devient $Q=\{(X_3,X_2)\}$

g. La valeur 1 de $D(X_3)$ n'ayant pas de support dans $D(X_1)$, on la supprime de $D(X_3)$ et on rajoute l'arc (X_2,X_3) à la file : $Q=\{(X_3,X_2),(X_2,X_3)\}$ et ci-dessous l'évolution des domaines :

X_i	X_1	X_2	X_3
$D(X_i)$	{1}	{2}	{2}

h. On prend l'arc (X_2,X_3) pour propagation (pour diversifier, je décide ici, AC3 ne me l'interdit pas, de prendre pour propagation l'arc que je viens à l'instant de rajouter à la file): Q devient $Q=\{(X_3,X_2)\}$

i. La valeur 2 de $D(X_2)$ n'ayant pas de support dans $D(X_3)$, on la supprime de $D(X_2)$ et ce dernier devient vide : échec et retour arrière pour essayer la deuxième valeur de X_1, la valeur 2 (fils droit de la racine) : le même raisonnement montre que le filtrage par AC3 au niveau du fils droit aboutit à un échec : échec et retour arrière. On se retrouve alors au niveau de la racine avec toutes les valeurs de X_1 déjà essayées : Look Ahead a bel et bien détecté l'inconsistance du CSP de départ

3) **Représentation matricielle de P :** $M_P=\begin{bmatrix} I2 & m & m \\ m & I2 & m \\ m & m & I2 \end{bmatrix}$, avec I2 la matrice identité 2x2 et m le poids commun des arcs de la représentation graphique (question 1). Il convient de noter ici que la matrice m est symétrique : elle est sa propre transposée.
4) L'algorithme de consistance de chemin PC2 détectera la triviale inconsistance de P de la façon suivante. Sa file sera initialisée aux arcs sur lesquels il y a une contrainte (dans un seul sens) : $Q=\{(X_1,X_2),(X_1,X_3),(X_2,X_3)\}$. Je prends l'arc (X_1,X_2) pour propagation, et je fais varier k de 1 à 3 : pour chaque valeur de k, je regarde si la modification qu'a subie à un moment donné l'arc que je propage, ici (X_1,X_2), peut avoir des répercussions sur les voisins (X_1,X_k) et (X_k,X_2). Puisqu'il s'agit ici de montrer que PC2 détecte l'inconsistance, il faut savoir éviter les détours et aller droit au but, idéalement par le plus court chemin, ce que je fais ici en décidant de commencer par k=3, et le voisin (X_1,X_3) (rien n'interdit à PC2 de tirer aléatoirement la valeur de k à prendre, et le voisin à considérer) : l'opération de consistance de chemin que PC2 doit faire est $M_P[1,3]=M_P[1,3]\cap M_P[1,2]\circ M_P[2,2]\circ M_P[2,3]= m\cap m\circ I2\circ m$. Le calcul $m\circ I2$ étant égal à m, l'opération devient : $M_P[1,3]=m\cap m\circ m=\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\cap\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\circ\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}=\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\cap\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}=\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$. La file de PC2 est encore loin d'être vide mais comme une entrée de la représentation matricielle, ici l'entrée [1,3], vient de passer à la matrice zéro, PC2 prend fin en ayant détecté l'inconsistance de P.

Exercice 2 (7 points=2+1,5+1,5+2) :

On considère la description qualitative suivante de la scène spatiale composée des trois villes Alger, Béjaia et Oran, et d'un bâteau :

- Alger est à l'Ouest de Béjaia
- Alger est à l'Est d'Oran
- Le bâteau est à l'Ouest ou au Nord-Ouest d'Alger
- Le bâteau est au Nord-Est ou au Sud-Est de Béjaia

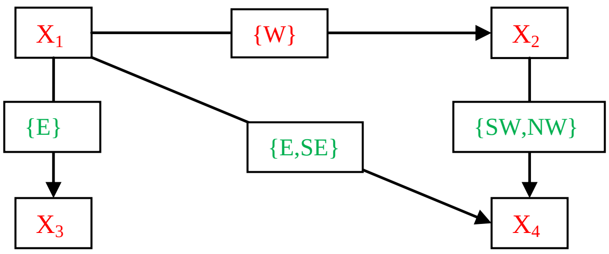
1) Modélisez la description à l'aide d'un CSP qualitatif P=(X,C) de directions cardinales.
2) Donnez la représentation graphique de P.
3) Donnez la représentation matricielle de P.
4) Utilisez l'algorithme de recherche Look-Ahead pour montrer l'inconsistance de P.

Solution :

1) **Modélisation à l'aide d'un CSP de directions cardinales P=(X,C) :**

   $X=\{X_1,X_2,X_3,X_4\}$, avec X_1,X_2,X_3 et X_4 variables désignant, respectivement, les positions d'Alger, de Béjaia, d'Oran et du bâteau. L'ensemble C des contraintes est comme suit :

   $C=\{c_1 : \{E\}(X_2,X_1),\ c_2 : \{W\}(X_3,X_1),\ c_3 : \{W,NW\}(X_4,X_1),\ c_4 : \{NE,SE\}(X_4,X_2)\}$
2) **Représentation graphique de P :**



3) **Représentation matricielle de P :**

   M_P=

	X_1	X_2	X_3	X_4
X_1	{EQ}	{W}	{E}	{E,SE}
X_2	{E}	{EQ}	?	{SW,NW}
X_3	{W}	?	{EQ}	?
X_4	{W,NW}	{NE,SE}	?	{EQ}

   avec ? désignant la relation universelle de l'algèbre des directions cardinales, et qui consiste en l'ensemble des neuf relations atomiques de l'algèbre : ?={EQ,N,NE,E,SE,S,SW,W,NW}

4) **Détection de l'inconsistance avec l'algorithme de recherche Look-Ahead :**

   L'algorithme Look-Ahead commence par le tout premier filtrage par PC2 au niveau de la racine de l'arbre de recherche : filtrage du CSP de départ (prétraitement, avant toute instanciation de paires de variables). La file de PC2 pour le prétraitement est initialisée aux paires de variables sur lesquelles il y a une contrainte : Q={(X1,X2),(X1,X3),(X1,X4),(X2,X4)}. On prend l'arc (X1,X2) pour propagation : k varie de 3 à 4. On commence par k=4 et le voisin (X1,X4) de (X1,X2) ; et on fait l'opération de consistance de chemin suivante : $M_P[1,4]=M_P[1,4]\cap M_P[1,2]°M_P[2,2]°M_P[2,4]$={E,SE}∩{W}°{EQ}°{SW,NW}. Le calcul {W}°{EQ}°{ étant égal à {W}, l'opération devient : $M_P[1,4]$={E,SE}∩{W}°{SW,NW}={E,SE}∩{SW,NW}={}

   Conclusion : le prétraitement de Look-Ahead (le tout premier filtrage, au niveau de la racine) détecte l'inconsistance de P

Exercice 3 (5 points=2,5+2,5) :

1) Ecrire un programme Prolog saisissant le CSP de l'exercice 1, et faisant appel au solveur sur domaines finis pour sa résolution.
2) Ecrire un programme Prolog utilisant le solveur sur domaines finis pour calculer toutes les listes d'entiers de longueur N, dont les éléments sont dans l'ensemble {1,…,M}, et tels que l'élément i est différent de l'élément (i+1), pour tout i∈{1,…,N-1}.

Solution :

```
:- use_module(library('clp/bounds')).
/* Question 1) */
cspexo1(Vars) :-
        /* on déclare les variables */
        length(Vars,3),
        /* le domaine de chaque variable est {1,2} */
        Vars in 1..2,
        /* Les sommets du triangle doivent prendre des couleurs différentes */
        all_different(Vars),
        /* on recherche une affectation des variables qui soit solution */
        label(Vars).
/* Question 2) */
listesNM(N,M,Vars) :-
        /* on déclare les variables : c'est une liste de N éléments */
        length(Vars,N),
        /* le domaine de chaque variable est {1,2,...,M} */
        Vars in 1..M,
```

```
        /* Un élément et son successeur immédiat doivent être distincts */
        contraintes(Vars),
        /* on recherche une affectation des variables qui soit solution */
        label(Vars).
contraintes([_]) :- !.
contraintes([XI,XIp1|Vars]) :-
        XI #\= XIp1,
        contraintes([XIp1|Vars]).
```