Cours de programmation par contraintes

Travaux Pratiques de modélisation

<u>Préambule :</u> Installer l'interface MiniZinc depuis www.minizinc.org sur sa machine, et avoir le tutorial dans un butineur web.

Include / paramètres / variables / contraintes / solve / output

include « alldifferent.mzn » ;

Paramètres :   int : n = 6;   array[1..3] of int : vecteur = [1, 2, 3] ;

Variables de décision : var 1..10 : x;   array[1..n] of var 1..10 : vecteur;   array[1..10, 1..5] of var 1..10 : tableau ;

Contraintes :   constraint x < y ;   constraint a < b ∧ b < c;   constraint a = 1 ∨ a = 2 ;

forall(i in 1..10) (x[i] < y[i]) ;   forall(i, j in 1..n where i < j) (x[i] < y[j]);

constraint sum(i in 1..10) (x[i]) = sum ;

solve satisfy;   solve minimize X;   solve maximize Y;

output [« x = » ++ show(x) ++ « \n » ++ « y = » ++ show(y)] ;

<u>Exercice 2 : cryptarithmes.</u> Ecrire deux modèles qui résolvent le cryptarithme présenté en cours : SEND + MORE = MONEY.

Exercice optionnel : résoudre le même problème mais à la main, avec un stylo et une feuille de papier.

Modéliser et résoudre le 2e cryptarithme présenté en cours : UN + DEUX + DEUX + DEUX + DEUX = NEUF

Exercice optionnel : Regarder sur Wikipedia à « cryptarithme » et modéliser et résoudre tous les problèmes posés.

<u>Exercice 3 : les N-reines.</u> Ecrire le modèle à deux coordonnées qui résout le problème des N-reines. Tester pour n = 8. Puis pour n = 10, 50, 100, 500, 1000, 10000. Que constate-t-on ? Pourquoi ?

Programmer le même problème, mais selon le modèle des colonnes. Augmenter le nombre de reines, comme avant. Que constate-t-on, en comparaison avec le premier modèle ? Pourquoi ?

<u>Exercice 4 : coloration de graphe.</u> Ecrire un modèle qui colorie chaque pays en une couleur différente des pays voisins. Tester avec le cas particulier : France, Belgique, Luxembourg, Hollande, Allemagne et Angleterre. (Regarder sur internet pour avoir le lien de juxtaposition entre les pays).

Exercice 5 : le problème du sac à dos. Un randonneur prépare son sac à dos (volume fixe) avant de partir en montagne. Il possède des boites de conserve, chacune possède un poids et un volume. Notre randonneur cherche à placer dans son sac le plus grand poids possible en boites de conserve, tout en ne dépassant pas le volume de son sac. Pourriez-vous lui dire quelles sont les boites qu'il peut emmener dans son sac à dos ? (On suppose qu'il y a 10 boites de conserve, de poids et de volume quelconques fixes).

Exercice 6 : le carré magique. Un carré magique de dimension N est un tableau NxN de nombres, tel que la somme des cases d'une ligne, la somme des cases d'une colonne, ou la somme des cases d'une des deux diagonales ont la même valeur. Trouver un carré magique 5x5 où la 2^{e} diagonale vaut 7, 19, 15, 6, 18.

Exercice 7 : l'usine chimique. Une usine chimique fabrique n produits différents à partir de m composants (par exemple, $n = 10$ et $m = 5$). Chaque produit requiert une certaine quantité de chaque composant. Chaque produit possède un certain prix unitaire. Chaque composant est disponible en stock dans une certaine quantité, qui ne peut pas être dépassée.

Pourriez-vous dire au directeur de cette usine comment maximiser son chiffre d'affaire en fonction du stock de composants dont il dispose ?

Exercice 8 : les gardiens corrompus. Un garçon veut offrir une pomme à une jeune fille. Pour l'atteindre, il doit passer par cinq portails, chacun avec un garde. Il corrompt chaque garde avec la moitié de ses pommes, plus une. Une fois qu'il a donné une pomme à la jeune fille, il n'a plus de pommes. Combien de pommes avait-il au départ ?

Exercice 9 : le sudoku. Modéliser le problème du sudoku 9x9 et résoudre les cas particuliers suivants (« 0 » signifie « chiffre à trouver ») :

1) Le sudoku vide (aucune case prédéfinie)

2)

8 0 0 1 0 0 0 0 0

0 5 0 9 0 7 2 0 0

0 1 9 3 0 2 0 0 7

1 0 0 0 0 0 0 5 6

5 6 0 4 0 8 0 7 9

3 7 0 0 0 0 0 0 4

7 0 0 5 0 9 4 1 0

0 0 5 8 0 1 0 6 0

0 0 0 0 0 4 0 0 3

3)

0 8 0 7 0 2 0 0 5

6 7 9 0 0 0 0 0 0

2 0 0 0 6 0 0 3 7

0 0 0 0 1 0 3 0 0

0 0 0 5 0 9 0 0 0

0 0 6 0 4 0 0 0 0

3 4 0 0 8 0 0 0 0

0 0 0 0 0 0 2 8 9

9 0 0 6 0 1 0 7 0

4)

0 0 0 0 0 0 2 0 0

9 0 2 0 0 5 0 3 0

6 0 0 1 4 0 0 7 0

8 0 1 0 7 0 5 0 2

0 0 0 6 0 0 1 9 0

0 0 0 0 0 0 0 0 0

0 1 0 0 0 0 0 0 0

0 0 6 3 1 7 0 0 0

0 7 5 0 6 0 0 0 0

5) http://www.mirror.co.uk/fun-games/sudoku/2010/08/19/world-s-hardest-sudoku-can-you-solve-dr-arto-inkala-s-puzzle-115875-22496946/

0 0 5 3 0 0 0 0 0

8 0 0 0 0 0 0 2 0

0 7 0 0 1 0 5 0 0

4 0 0 0 0 5 3 0 0

0 1 0 0 7 0 0 0 6

0 0 3 2 0 0 0 8 0

0 6 0 5 0 0 0 0 9

0 0 4 0 0 0 0 3 0

0 0 0 0 0 9 7 0 0

Exercice 10 : le nombre magique. Un nombre se représentant comme x_0 x_1 x_2 … x_{N-1} en base 10 est magique ssi pour tout *i* de *0* à *N-1*, *i* apparait x_i fois dans la représentation en base 10 de ce nombre. Trouver le nombre magique pour N = 4.

Exercice 11 : le problème de Richard Korf. Chercher les dimensions du rectangle de surface minimale qui contient n carrés 1x1 2x2 …. n x n ne se chevauchant pas.

Exercice 12 : le pont. L'énoncé en anglais est disponible à l'adresse suivante, avec le pont à construire en Fig. 1.

http://mozart2.org/mozart-v1/doc-1.4.0/fdt/node48.html#fig.bridgeproblem

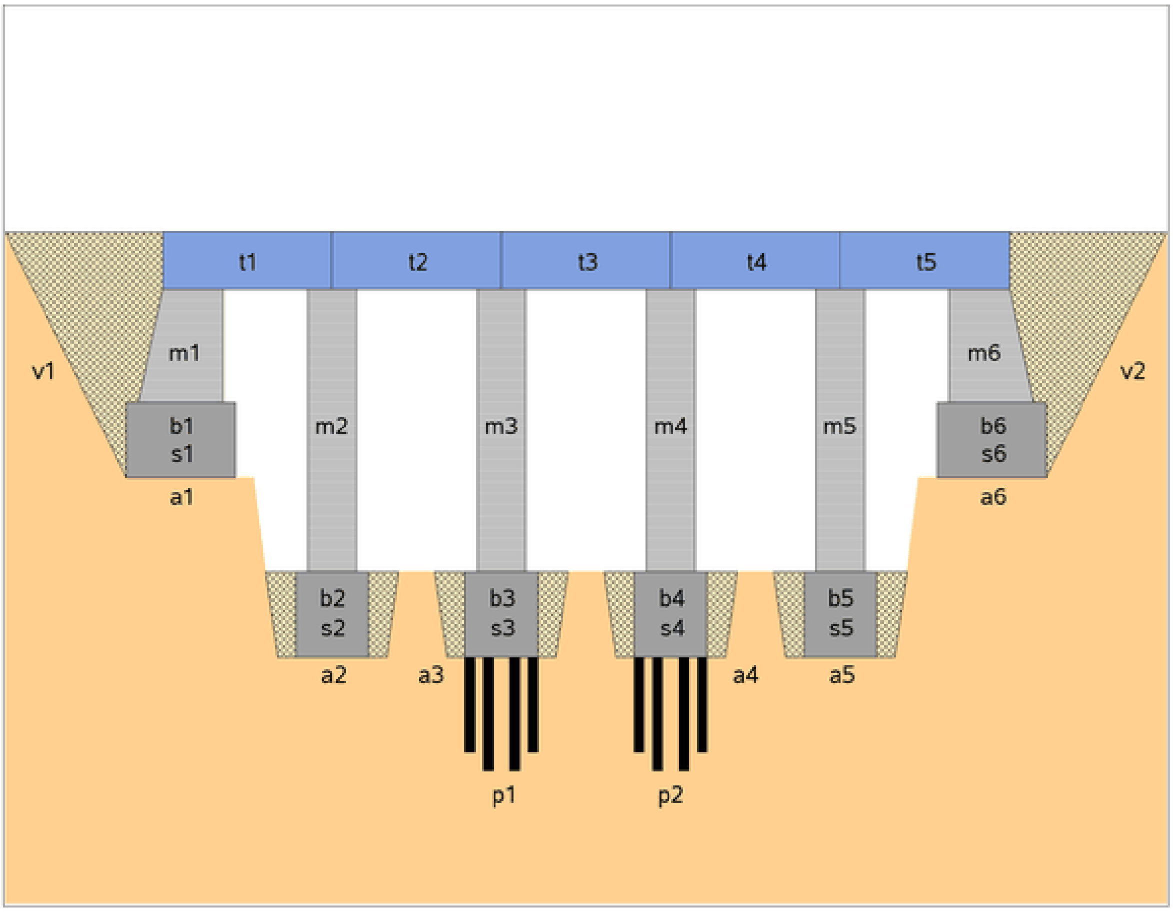

Figure 1 : le pont à construire.