Exercice 1

- On s’intéresse à un distributeur automatique de boissons. L'utilisateur insère des pièces de monnaie pour un total de T centimes d'Euros, puis il sélectionne une boisson, dont le prix est de P centimes d'Euros (T et P étant des multiples de 10).

  Il s'agit alors de calculer la monnaie à rendre, sachant que le distributeur a en réserve E2 pièces de 2 €, E1 pièces de 1€, C50 pièces de 50 centimes, C20 pièces de 20 centimes et C10 pièces de 10 centimes. Modélisez ce problème sous la forme d'un CSP.

- Comment pourrait-on exprimer le fait que l'on souhaite que le distributeur rende le moins de pièces possibles ?

Exercice 1 : solution

On définit le CSP *(X,D,C)* tel que :

- $X = \{XE2, XE1, XC50, XC20, XC10\}$

où *XE2* désigne le nombre de pièces de 2 Euros à retourner, *XE1* désigne le nombre de pièces de 1 Euro à retourner, …

- Les domaines spécifient que la quantité de pièces retournées, pour un type de pièce donné, est comprise entre 0 et le nombre de pièces de ce type que l'on a en réserve :

$$D(XE2) = \{0,1,...,E2\}$$
$$D(XE1) = \{0,1,...,E1\}$$
$$D(XC50) = \{0,1,...,C50\}$$
$$D(XC20) = \{0,1,...,C20\}$$
$$D(XC10) = \{0,1,...,C10\}$$

Exercice 1 : solution

- Les contraintes spécifient que la somme à retourner doit être égale à la somme insérée moins le prix à payer :

$C = \{ 200*XE2 + 100*XE1 + 50*XC50 + 20*XC20 + 10*XC10 = T-P \}$

Dans cette modélisation, nous avons utilisé une contrainte arithmétique linéaire sur les entiers.

Pour exprimer le fait que l'on souhaite que le distributeur rende le moins de pièces possibles, on pourrait ajouter à ce CSP une fonction "objectif" à minimiser :

$f(X) = XE2 + XE1 + XC50 + XC20 + XC10$

Dans ce cas, résoudre le CSP revient à chercher une affectation de *X* complète et consistante qui minimise *f(X)*.

Exercice 2: Cryptarithme

- On considère l'addition suivante :

- SEND

  \+ MORE

= MONEY

où chaque lettre représente un chiffre diffèrent (compris entre 0 et 9). On souhaite connaître la valeur de chaque lettre, sachant que la première lettre de chaque mot représente un chiffre diffèrent de 0.

- Modélisez ce problème sous la forme d'un CSP.

Exercice 2: Solution

Première modélisation

•Variables : on associe une variable à chaque lettre $X = \{S,E,N,D,M,O,R,Y\}$

•Domaines : les variables correspondant au premier chiffre d'un mot (*S* et *M*) doivent prendre une valeur différente de 0 ; les autres peuvent prendre n'importe quelle valeur entre 0 et 9.
$D(S) = D(M) = \{1,2,3,4,5,6,7,8,9\}$
$D(E) = D(N) = D(D) = D(O) = D(R) = D(Y) = \{0,1,2,3,4,5,6,7,8,9\}$

•Contraintes :
•Une première contrainte exprime le fait que SEND+MORE=MONEY :
$C1 = 1000*S + 100*E + 10*N + D + 1000*M + 100*O + 10*R + E$
$= 10000*M + 1000*O + 100*N + 10*E + Y$

•Une seconde contrainte exprime le fait que toutes les variables doivent prendre des valeurs différentes. On peut utiliser pour cela la contrainte globale "toutes-différentes" :C2 = toutesdifferentes({S,E,N,D,M,O,R,Y})

Exercice 2: Solution

Deuxième modélisation

Une deuxième modélisation consiste à poser les contraintes "verticalement", comme quand on fait une addition à la main. Pour cela, on rajoute 3 variables *R1*, *R2* et *R3* correspondant aux retenues successives.

On obtient le CSP suivant :

•Variables :X = {S,E,N,D,M,O,R,Y,R1,R2,R3}

•Domaines :
D(S) = D(M) = {1,2,3,4,5,6,7,8,9}
D(E) = D(N) = D(D) = D(O) = D(R) = D(Y) ={0,1,2,3,4,5,6,7,8,9}
D(R1) = D(R2) = D(R3) = {0,1}

Exercice 2: Solution

•**Contraintes :**

•Un premier ensemble de contraintes exprime le fait que SEND+MORE=MONEY :

$$C1 = \{ D + E = Y + 10*R1,$$
$$R1 + N + R = E + 10*R2,$$
$$R2 + E + O = N + 10*R3,$$
$$R3 + S + M = O + 10*M \}$$

•Une dernière contrainte exprime le fait que toutes les variables doivent prendre des valeurs différentes. On peut utiliser pour cela la contrainte globale "toutes-différentes" **:**
$C2 = toutesdifferentes(\{S,E,N,D,M,O,R,Y\})$

Exercice 3: Le zèbre

Cinq maisons consécutives, de couleurs différentes, sont habitées par des hommes de différentes nationalités. Chacun possède un animal différent, a une boisson préférée différente et fume des cigarettes différentes. De plus, on sait que :

Le norvégien habite la première maison,
La maison à coté de celle du norvégien est bleue,
L'habitant de la troisième maison boit du lait,
L'anglais habite la maison rouge,
L'habitant de la maison verte boit du café,
L'habitant de la maison jaune fume des kools,
La maison blanche se trouve juste après la verte,
L'espagnol a un chien,
L'ukrainien boit du thé,

Exercice 3: Le zèbre

Le japonais fume des cravens,
Le fumeur de old golds a un escargot,
Le fumeur de gitanes boit du coca,
Le voisin du fumeur de Chesterfields a un renard,
Le voisin du fumeur de kools a un cheval.

Qui boit de l'eau ? A qui appartient le zèbre ?

Modélisez ce problème sous la forme d'un CSP.

Exercice 3: Solution

Variables du problème: on associe une variable par attribut (couleur, animal, boisson, nationalité, cigarette)

X = {blanche, rouge, verte, jaune, bleue, norvégien, anglais, ukrainien, japonais, espagnol, cheval, renard, zèbre, escargot, chien, thé, eau, lait, café, vin, kools, chesterfields, old_golds, cravens, gitanes}

Domaines des variables:
D(Xi) = {1,2,3,4,5}, pour toute variable Xi de X

Contraintes:
On pose tout d'abord une contrainte pour chaque assertion de l'énoncé :

Exercice 3: Solution

norvégien = 1,
bleue = norvégien + 1,
lait = 3,
anglais = rouge,
verte = café,
jaune = kools,
blanche = verte + 1,
espagnol = chien,
ukrainien = thé,
japonais = cravens,
old_golds = escargot,
gitanes = vin,
(chesterfields = renard + 1) ou (chesterfields = renard - 1),
(kools = cheval + 1) ou (kools = cheval - 1)

Exercice 3: Solution

De plus, toutes les variables de même "type" doivent avoir des valeurs différentes (il ne peut pas y avoir plusieurs maisons qui ont la même couleur, ou un même animal, …)

blanche ≠ rouge ≠ verte ≠ jaune ≠ bleue,
thé ≠ eau ≠ lait ≠ café ≠ vin,
norvégien ≠ anglais ≠ ukrainien ≠ japonais ≠ espagnol,
cheval ≠ renard ≠ zèbre ≠ escargot ≠ chien,
kools ≠ chesterfields ≠ old_golds ≠ cravens ≠ gitanes

Exercice 4: carré magique

Un carré magique de taille 3 est une matrice 3�3 composée des chiffres de 1 à 9 disposées de telle sorte que la somme de chaque ligne et de chaque colonne donne le même résultat.

Exemple :

8	3	4
1	5	9
6	7	2

Modéliser ce problème sous forme d’un CSP.

Exercice 4: Solution

Variables = {

X1,1, X1,2, X1,3,

X2,1, X2,2, X2,3,

X3,1, X3,2, X3,3

}

avec dom(xi,j) = {1, . . . , 9}, ∀i, j ∈ 1..9

Contraintes = {

$\sum_{j=1}^{3} X_{1,j} =$

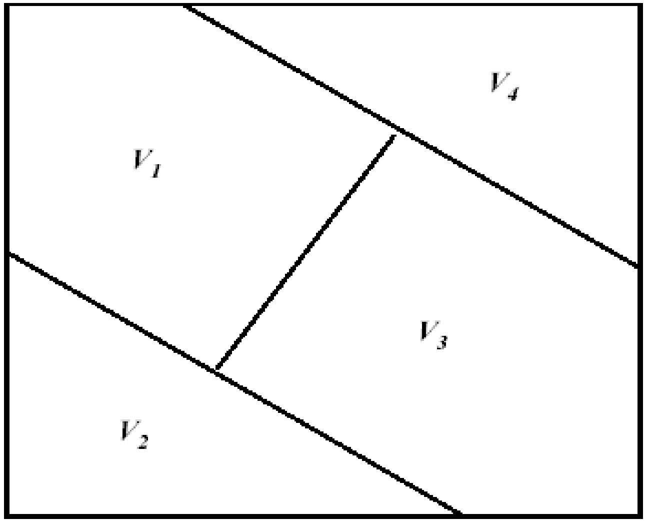
$\sum_{j=1} X_{2,j} =$

. . .

}

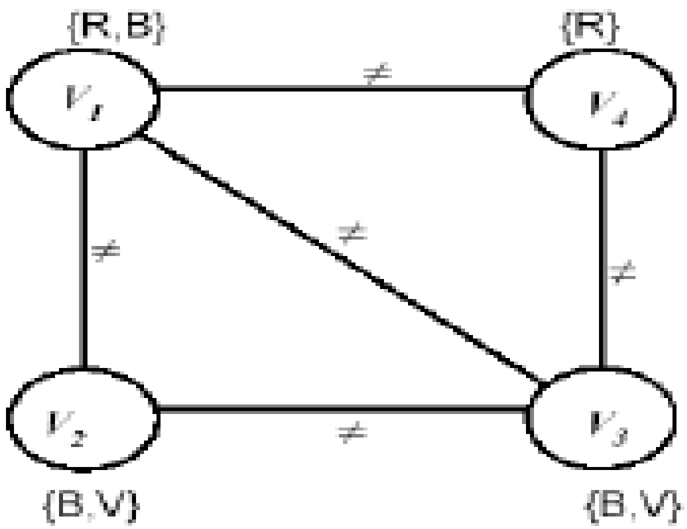
Exercice 5:

Soit la carte suivante décrivant les frontières entre quatre villes (V_1, ..., V_4). On voudrait colorier la carte en utilisant seulement les couleurs rouge, bleu et vert, de sorte que V_1 soit en rouge ou en bleu, V_2 et V_3 soient en bleu ou en vert et V_4 soit en rouge. Toutefois, deux villes adjacentes ne peuvent pas avoir la même couleur.



Exercice 5:

1. Donnez le graphe des contraintes modélisant ce problème comme un problème de satisfaction de contraintes. Vous devez clairement indiquer les domaines des variables et les contraintes entre ces dernières.



Exercice 5:

2. Donnez le résultat de l'algorithme AC-3 sur ce problème. Que concluez-vous de ce résultat?

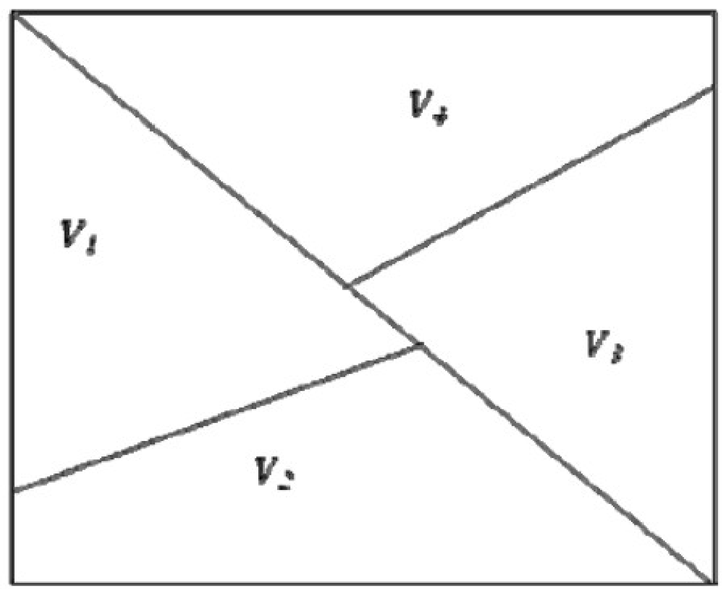
V_1	*V_2*	*V_3*	*V_4*	*Arc-consistency*
{R, B}	{B,V}	{B,V}	{R}	
{B}	{B,V}	{B,V}	{R}	$V_1 - V_4$
{B}	{V}	{B,V}	{R}	$V_2 - V_1$
{B}	{V}	{B}	{R}	$V_3 - V_2$
{B}	{V}	{}	{R}	$V_3 - V_1$

Conclusion :

Le domaine d'une des variables (V_3) devient vide. Il n'y a donc pas de solution.

Exercice 5:

Soit la carte suivante décrivant les frontières entres quatre villes (V_1, ..., V_4). On voudrait colorier la carte en utilisant seulement les couleurs rouge, bleu et vert, de sorte que V_1 soit en rouge ou en vert, V_2 et V_3 soient en bleu ou en vert; et V_4 soit en vert. Toutefois, deux villes adjacentes ne peuvent avoir la même couleur. Donnez le résultat de l'algorithme AC-3 sur ce problème. Que concluez-vous de ce résultat?



Exercice 5:

V_1	V_2	V_3	V_4	*AC-3*
{R, V}	{B,V}	{B,V}	{V}	
{R}	{B,V}	{B,V}	{V}	$V_1 - V_4$
{R}	{B,V}	{B }	{V}	$V_3 - V_4$
{R}	{V}	{B}	{V}	$V_2 - V_3$

Conclusion :

Le domaine de chaque variable est réduit exactement à une seule valeur. Comme il n'y a que des contraintes binaires, on a une solution.

Exercice 6:

Supposons que vous devez concevoir un menu pour une occasion spéciale. Vous prévoyez servir une Entrée (***E***), Boisson (***B***), Plat principal (***P***) et Dessert (***D***). Les choix possibles pour chacune des ces catégories sont :

E : salade (***s***), terrine de gibier (***t***);

B : eau (***e***), lait (***l***), nectar (***n***);

P: poisson (***p***), bœuf (***b***), spaghettis maison (***m***);

D: fruits tropicaux (***f***); crème glacée (***c***), gâteau au fromage (***g***).

Étant donné que vous aurez un seul menu pour tous les invités, il doit satisfaire les contraintes suivantes.

Exercice 6:

C1 : Afin que les végétariens puissent manger, si l'entrée est une terrine de gibier alors le plat principal doit être le poisson ou les spaghettis maison.

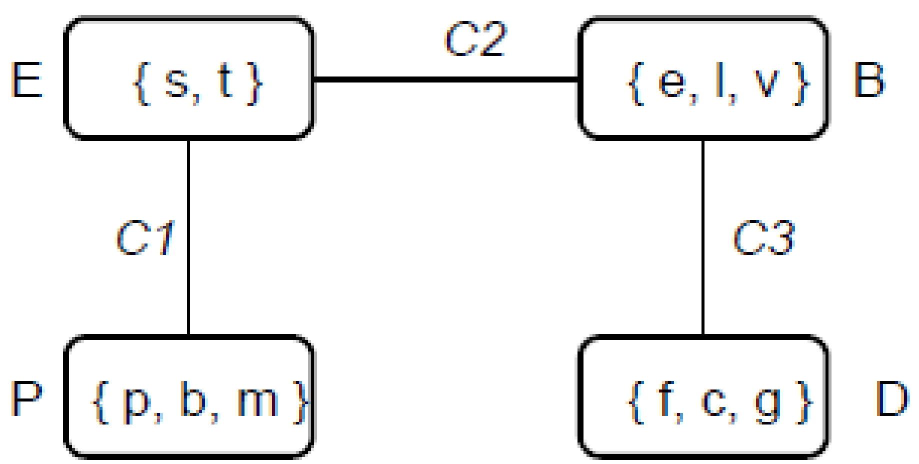
C2 : Étant donné votre budget limité, si vous servez la terrine de gibier, alors vous ne pourrez pas vous payer le cocktail, ni même le lait.

C3 : Pour avoir un apport en calcium, vous devrez servir au moins l'un des éléments suivants : le lait, la crème glacée ou le gâteau au fromage.

Exercice 6:

Considérons ce problème comme un problème **CSP**, en posant ***E, B,*** ***P***, et ***D*** comme variables. Le domaine de chaque variable correspond alors aux choix possibles correspondants tels qu'indiqués plus haut.

1, Dessinez le graphe de contraintes du problème. Prenez soin d'indiquer les domaines de valeurs des variables et les contraintes.



Exercice 6:

2, Supposons que l'on assigne d'abord ***t*** à la variable ***E***, comme première étape de *backtracking search* et qu'on applique ensuite *forward-checking*. Indiquez le domaine de valeurs de chaque variable à la fin de *forward-checking*.

E = {t}

B = {e} - l et v éliminés à cause de la contrainte *C2*

P = {p, m} - b éliminé à cause de la contrainte *C1*

D = {f, c, g}

Exercice 6:

3, Supposons encore une fois que l'on assigne d'abord ***t*** à la variable ***E***, comme première étape de *backtracking-search* et qu'on applique ensuite *AC-3*. Indiquez le domaine de valeurs de chaque variable à la fin d'*AC-3*.

E = {t}

B = {e} - l et v éliminés à cause de la contrainte *C2* pour l'arc (E, B)

P = {p, m} - l et v éliminés à cause de la contrainte *C1* pour l'arc (E, P)

D = {c, g} - l et v éliminés à cause de la contrainte *C3* pour l'arc (P, D)

Exercice 6:

4,   Est-il vrai qu'AC-3 *coupera* toujours au moins autant de valeurs des domaines des variables que *forward-checking* ? Répondez d'abord par « oui » ou « non » et ensuite expliquez brièvement votre réponse.

Oui

La première étape d'AC-3 est équivalente au *forward-checking*. Puis elle propage les contraintes directement affectées par l'assignation de la valeur à la variable courante.