

Programmation Par Contraintes

Présentation générale

David Savourey

CNRS, École Polytechnique

inspiré des cours de Philippe Baptiste et Ruslan Sadykov

- ① Généralités sur la PPC
- ② PPC : Définitions
- ③ Modélisations en CSP
- ④ Binarisation
- ⑤ Méthodes de résolution
- ⑥ Exercices

- 2 intervenants : David Savourey, Éric Nespoulous (IBM)
- Séance 1 : Généralités (DS)
- Séance 2 : Consistances (DS)
- Séance 3 : Solveur PPC IBM (EN)
- Séance 4 : Apprentissage (DS)
- Séance 5 : Étude d'un article (DS)
- Séance 6 : Soutenances de projet (DS)

Généralités sur la PPC

- Problème de décision : la réponse est “oui” ou “non”
- Problème d'optimisation : minimiser ou maximiser une certaine fonction objectif
- PPC naturellement faite pour la décision

- trouver une solution (décision)
- trouver toutes les solutions
- trouver une solution optimale (optimisation)

- ex : trouver un plus court chemin
- on résout successivement plusieurs problèmes de décision (un chemin de moins de 60 ? de 58 ? de 40 ? de 52 ? de 55 ? etc.)
- en général par dichotomie

- Programmation mathématique :
 - linéaire
 - linéaire en nombres entiers
 - quadratique
 - etc.
- Méta-heuristiques :
 - recherche locale
 - algos génétiques
 - méthodes tabous
 - recuit simulé
 - etc.
- Programmation Par Contraintes

- on travaille sur un problème de décision
- le langage d'expression est très peu contraint
- on se sert des contraintes pour accélérer la résolution

PPC : Définitions

- En PPC, on cherche à résoudre un CSP. Ce dernier est donné par le triplet $\langle X, D, C \rangle$ où :
 - X est un ensemble de variables x_1, x_2, \dots, x_n ;
 - D est un ensemble de domaines D_1, D_2, \dots, D_n ;
 - C est un ensemble de contraintes C_1, C_2, \dots, C_m .
- Le domaine D_i est l'ensemble des valeurs possibles pour la variable x_i .
- Chaque contrainte C_j se définit par :
 - son arité : le nombre de variable sur lesquelles elle porte ;
 - la liste de ces variables ;
 - l'ensemble des tuples qui la satisfont.

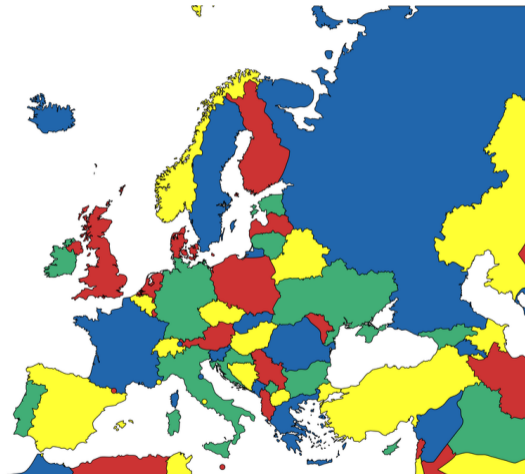
- variables : A, B, C, D
- domaines :
 - $D_A = \{1, 4, 5, 8\}$
 - $D_B = \{2, 3, 5, 7, 9\}$
 - $D_C = \{4, 8, 7, 9\}$
 - $D_D = \{1, 4, 5, 7, 8, 9\}$
- contraintes :
 - $C_1(A, C) : \{(1, 7), (1, 9), (5, 9)\}$
 - $C_2(A, D) : \{(1, 1), (1, 5), (5, 5), (5, 9), (8, 9)\}$
 - $C_3(C, D) : \{(4, 1), (8, 1), (9, 7)\}$
 - $C_4(B, D) : \{(2, 7), (2, 9), (5, 8), (7, 9), (9, 9)\}$
- Ce CSP est binaire : toutes les contraintes portent sur exactement 2 variables.

- Il est toujours possible de coder les contraintes “en extension”, c'est-à-dire par un ensemble de tuples.
- Les solveurs de PPC utilisent par défaut un codage en extension.
- Définir une contrainte en intension veut dire utiliser des opérateurs dont la sémantique est connue.
- Par exemple, si $D_x = [1, 5]$ et $D_y = [2, 4]$, alors la contrainte en intension “ $x \neq y$ ” est équivalente à $\{(1, 2), (1, 3), (1, 4), (2, 3), (2, 4), (3, 2), (3, 4), (4, 2), (4, 3), (5, 2), (5, 3), (5, 4)\}$.

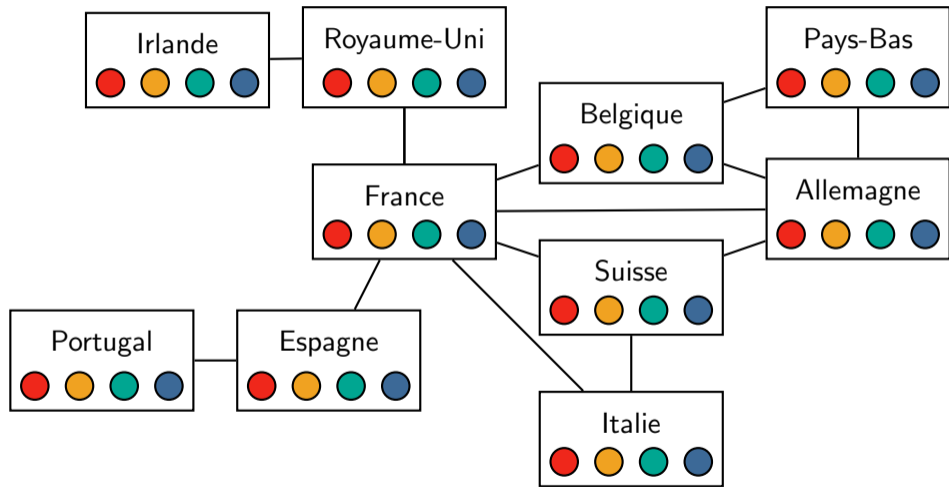
- Une instanciation est une affectation complète de valeurs aux variables. Par exemple, pour le CSP précédent, $\{\langle A, 1 \rangle, \langle B, 7 \rangle, \langle C, 4 \rangle, \langle D, 9 \rangle\}$.
- Une instanciation est une solution valide si les valeurs données aux variables sont telles que toutes les contraintes sont vérifiées.

Modélisations en CSP

- graphe planaire
- 2 voisins doivent avoir des couleurs différentes
- trouver une solution avec 4 couleurs seulement



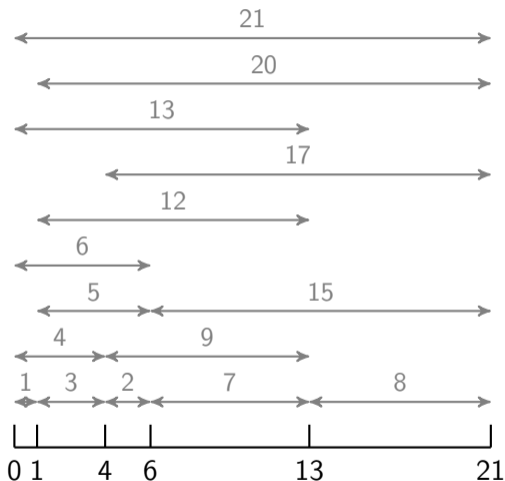
- une variable par pays
- toutes les variables ont pour domaine $\{R, B, J, V\}$.
- si 2 pays x et y sont voisins, on met la contrainte $x \neq y$.



- sur un carré de $n \times n$
- placer les nombres de 1 à n^2
- toutes les rangées de sommes égales
- quel modèle?

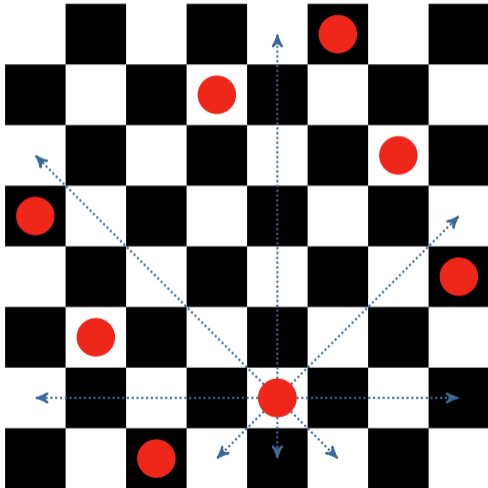
6	1	8	↗ =15
7	5	3	→ =15
2	9	4	→ =15
↓ =15	↓ =15	↓ =15	↘ =15

- x_1, x_2, \dots, x_{n^2}
- $D_1 = D_2 = D_{n^2} = [1, 2, \dots, n^2]$
- $\forall i \neq j \in [1, n^2]^2, x_i \neq x_j$
- $\forall r \in \{L, C, D\}, \sum_{x_i \in r} x_i = n(n^2 + 1)/2$



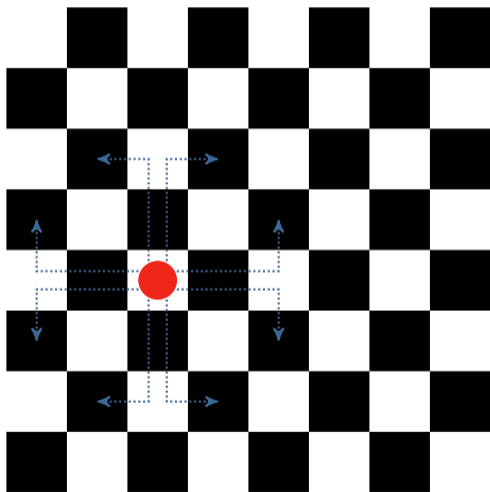
- fabriquer une règle avec n marques
- toutes les distances entre les marques doivent être différentes
- version optim : en minimisant la valeur de la plus grande marque.

- variable x_i : valeur de la i ème plus petite marque
- x_1, x_2, \dots, x_n
- $D_1, D_2, \dots, D_n = [1, UB(n)]$
- $x_1 = 1$
- $\forall i \in [2, n], x_i > x_{i-1}$
- $\forall (i < j) \neq (k < l), x_j - x_i \neq x_l - x_k$



- sur une grille de $n \times n$ cases
- placer n reines
- aucune reine ne doit pouvoir manger une autre reine

- variable c_i : indice de la colonne occupée par la reine qui se trouve sur la i ème ligne
- c_1, c_2, \dots, c_n
- $D_1, D_2, \dots, D_n = [1, n]$
- $\forall i \neq j \in [1, n]^2, c_i \neq c_j$
- $\forall i < j \in [1, n]^2, c_j - c_i \neq j - i$
- $\forall i < j \in [1, n]^2, c_i - c_j \neq j - i$



Le problème dit du cavalier d'Euler peut s'énoncer comme suit : Comment faire visiter par un cavalier chaque case d'un échiquier pour le faire revenir à son point de départ sans jamais repasser sur la même case ?

- on numérote les cases de 1 à n
- on note p_i la position du cavalier à la i ème étape
- p_1, p_2, \dots, p_n
- $D_1, D_2, \dots, D_n = [1, n]$
- $\forall i \neq j \in [1, n]^2, p_i \neq p_j$
- $\forall i \in [2, n], p_i \in V(p_{i-1})$
- $p_n \in V(p_1)$
- La relation $x \in V(y)$ est décrite en extension :

$$\{(1, 11), (1, 18), (2, 12), (2, 17), (2, 19), \dots\}$$

Binarisation

On dira qu'un CSP est binaire lorsque toutes les contraintes sont d'arité 2.

On dira qu'un CSP est booléen lorsque tous les domaines des variables sont $\{0, 1\}$.

- on peut toujours binariser un CSP
- la plupart des techniques développées travaillent sur des CSP binaires
- dans la suite, on considèrera des CSP binaires uniquement

Soit le CSP suivant :

- A, B, C, D
- $D_A = D_B = D_C = D_D = \{0, 1, 2\}$
- $A + B + C = 2, B + C + D = 5$

Version binaire équivalente :

- A, B, C, D, X, Y
- $D_A = D_B = D_C = D_D = \{0, 1, 2\}, D_X = \{011, 101, 110, 002, 020, 200\}$ et $D_Y = \{122, 221, 212\}$
- $C(A, X) = \{(0, 011), (0, 002), (0, 020), (1, 101), (1, 110), (2, 200)\}$
 $C(B, X) = \{(1, 011), (0, 002), (2, 020), (0, 101), (1, 110), (0, 200)\}$
 $C(C, X) = \{(1, 011), (2, 002), (0, 020), (1, 101), (0, 110), (0, 200)\}$
 $C(B, Y) = \{(1, 122), (2, 221), (2, 212)\}$ $C(C, Y) = \{(2, 122), (2, 221), (1, 212)\}$
 $C(D, Y) = \{(2, 122), (1, 221), (2, 212)\}$

Soit le CSP suivant :

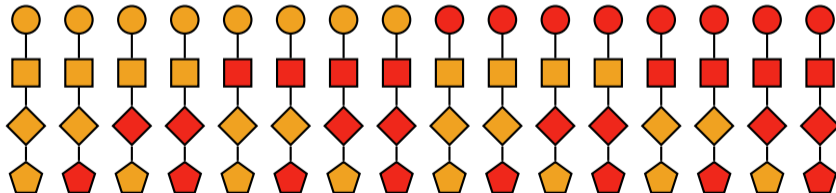
- A, B, C, D
- $D_A = D_B = D_C = D_D = \{0, 1, 2\}$
- $A + B + C = 2, B + C + D = 5$

Version binaire équivalente :

- $X_1, X_2,$
 $D_1 = [002, 011, 020, 101, 110, 200] \quad A + B + C = 2$
 $D_2 = [122, 212, 221] \quad B + C + D = 5$
- $C(X_1, X_2) = [(011, 122), (110, 122), (020, 212), (020, 221)] \quad B$
- $C(X_1, X_2) = [(002, 122), (002, 221), (011, 212), (101, 212)] \quad C$

Méthodes de résolution

pour chaque instantiation complète i faire
| **si i respecte toutes les contraintes alors**
| | Retourner *VRAI* ;
Retourner *FAUX* ;



Données : Une instantiation partielle i

si i viole une contrainte alors

| Retourner *FAUX* ;

si i est complète alors

| Retourner *VRAI* ;

Choisir une variable x non instanciée ;

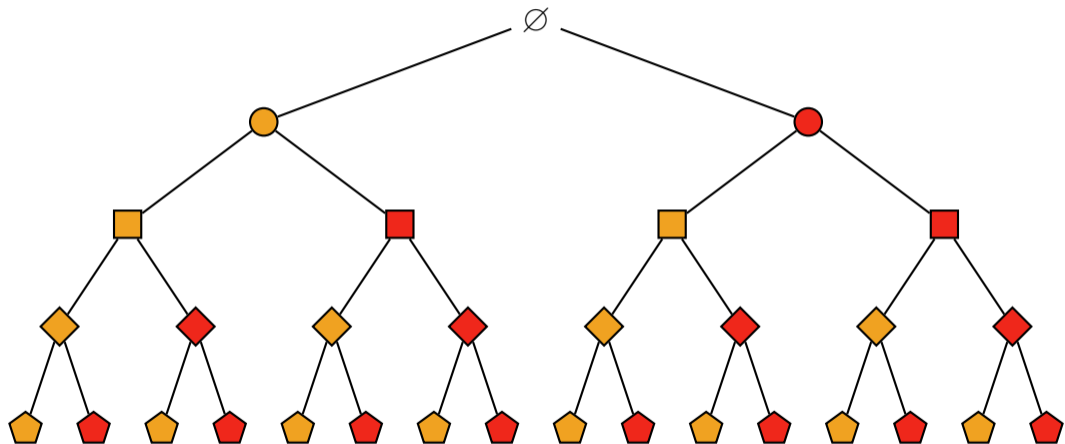
pour chaque valeur v dans D_x faire

| $j \leftarrow i \cup \langle x, v \rangle$;

| **si *Backtrack*(j) alors**

| | Retourner *VRAI* ;

Retourner *FAUX* ;



Exercices

```
  S E N D
+ M O R E
-----
M O N E Y
```

Deux modèles possibles

- chaque lettre correspond à un chiffre compris entre 0 et 9 différent
- les nombres ainsi formés ne peuvent pas commencer par 0

Modèle sans retenues :

- variables : S,E,N,D,M,O,Y,R
- domaines : [0,9]
- alldiff sur les variables
- $S \neq 0, M \neq 0$
- $1000S+100E+10N+D+1000M+100O+10R+E=10000M+1000O+100N+10E+Y$

Modèle avec retenues :

- variables : S,E,N,D,M,O,Y,R, r_1,r_2,r_3,r_4
- domaines lettres : [0,9], domaines retenues : [0,1]
- alldiff sur les lettres
- $D + E = Y + 10r_1$
- $r_1 + N + R = E + 10r_2$
- $r_2 + E + O = N + 10r_3$
- $r_3 + S + M = O + 10r_4$
- $r_4 = M$

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

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