

# Programmation par Contraintes

Module du Master "Systèmes Informatiques Intelligents" 2ème année

## Annexe 2-1

Résolution d'un CSP binaire discret : consistance de chemin

Mr ISLI

Faculté d'Informatique

Département Intelligence Artificielle et Science des Données

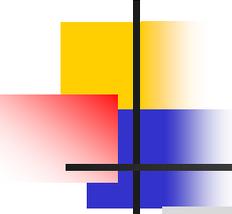
Université des Sciences et de la Technologie Houari Boumediène

BP 32, El-Alia, Bab Ezzouar

DZ-16111 ALGER

[https://perso.usthb.dz/~aisli/TA\\_PpC.htm](https://perso.usthb.dz/~aisli/TA_PpC.htm)

[amar.isli@usthb.edu.dz](mailto:amar.isli@usthb.edu.dz)



# Annexe 2-1

## Résolution d'un CSP binaire discret : consistance de chemin

---

### Introduction

Cette annexe présente les deux algorithmes de consistance de chemin PC-1 et PC-2 ; de même que l'algorithme Look\_Ahead avec PC-2 comme procédure de filtrage durant la recherche.

# Annexe 2-1

## Résolution d'un CSP binaire discret : consistance de chemin

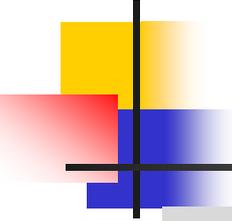
### Consistance de chemin (path-consistency)

Un CSP  $P=(X,D,C)$  est consistant de chemin si, pour tout chemin  $(X_{i0}, \dots, X_{ij})$  de longueur  $j$  :

$X_{i0}$	$X_{i1}$	...	$X_{ik}$	$X_{i(k+1)}$	...	$X_{ij}$
$v_0$	$v_1$	...	$v_k$	$v_{(k+1)}$	...	$v_j$

Pour tous  $(v_0, v_j)$  tels que unaires+binaires sur  $X_{i0}$  et  $X_{ij}$  satisfaites, il existe  $v_1, \dots, v_{(j-1)}$  tels que pour tout  $k=0\dots(j-1)$ :

$(v_k, v_{(k+1)})$  satisfait unaires+binaires sur  $X_{ik}$  et  $X_{i(k+1)}$



## Annexe 2-1

Résolution d'un CSP binaire discret : consistance de chemin

---

### consistance de chemin (path-consistency)

On montre facilement qu'on peut se restreindre aux chemins de longueur 2 (triplets de variables).

# Annexe 2-1

## Résolution d'un CSP binaire discret : consistance de chemin

### consistance de chemin (path-consistency)

Un CSP  $P=(X,D,C)$  est consistant de chemin si :

$X_{i0}$

$X_{i1}$

$X_{i2}$

$v_0$

$v_1$

$v_2$

Pour tous  $(v_0, v_2)$  tels que unaires+binaires sur  $X_{i0}$  et  $X_{i2}$  satisfaites, il existe  $v_1$  tel que :

$(v_0, v_1)$  satisfait unaires+binaires sur  $X_{i0}$  et  $X_{i1}$

$(v_1, v_2)$  satisfait unaires+binaires sur  $X_{i1}$  et  $X_{i2}$

$v_1$  support de la paire  $(v_0, v_2)$

# Annexe 2-1

## Résolution d'un CSP binaire discret : consistance de chemin

### Principe d'un algorithme de consistance de chemin

Filtrage des paires de valeurs permises :

- pour chaque paire  $(X_i, X_j)$  de variables
  - supprimer les paires permises  $(v_i, v_j)$  n'ayant pas de support dans le domaine d'une troisième variable  $X_k$

$X_i$

$X_k$

$X_j$

$v_0$

$v_1$

$v_2$

Supprimer la paire permise  $(v_0, v_2)$  si elle n'a aucune valeur  $v_1$  de  $D(X_k)$  comme support.

## Annexe 2-1

Résolution d'un CSP binaire discret : consistance de chemin

fonction REVISE\_pc(( $X_i, X_k, X_j$  ), ( $X, D, C$ )) : booléen

**début**

temp =  $M_p[i,j] \cap M_p[i,k] \circ M_p[k,k] \circ M_p[k,j]$

**si** temp  $\neq$   $M_p[i,j]$  **alors**

$M_p[i,j] =$  temp

    retourner VRAI

**sinon** retourner FAUX

**finsi**

**fin**

# Annexe 2-1

## Résolution d'un CSP binaire discret : consistance de chemin

fonction PC1((X,D,C))

Début

Q ←  $\{(X_i, X_k, X_j) \in X^3 : \neg(X_i = X_k = X_j)\}$

**répéter**

R ← FALSE

**pour** tous les  $(X_i, X_k, X_j)$  de Q **faire**

**si** REVISE\_pc( $(X_i, X_k, X_j), (X, D, C)$ ) **alors** R ← TRUE **finsi**

**fin pour**

**jusqu'à** non R

retourner (X,D,C)

fin

# Annexe 2-1

## Résolution d'un CSP binaire discret : consistance de chemin

```
procedure PC2( $M_p$ )           %version 1%
début
 $Q \leftarrow \{(X_i, X_k, X_j) \in X^3 : (i \leq j) \text{ et } \neg(i=k=j)\}$ ; entrée_nulle=false;
tant que ( $Q \neq \emptyset$  et  $\neg$ entrée_nulle) faire
  Prendre un triplet  $(X_i, X_k, X_j)$  de variables de  $Q$ , et l'en supprimer ( $Q \leftarrow Q \setminus \{(X_i, X_k, X_j)\}$ );
  temp= $M_p[i,j] \cap M_p[i,k] \circ M_p[k,k] \circ M_p[k,j]$ ;
  si temp=[0] alors retourner FAUX finsi
  si temp $\neq M_p[i,j]$  alors
     $M_p[i,j] = \text{temp}$ ;  $M_p[j,i] = (\text{temp})^t$ ;
     $Q = Q \cup \{(X_i, X_j, X_m) \in X^3 : (i \leq m) \text{ et } \neg(i=j=m)\} \cup \{(X_m, X_i, X_j) \in X^3 : (m \leq j) \text{ et } \neg(m=i=j)\}$ 
       $\cup \{(X_j, X_i, X_m) \in X^3 : (j \leq m) \text{ et } \neg(j=i=m)\} \cup \{(X_m, X_j, X_i) \in X^3 : (m \leq i) \text{ et } \neg(m=j=i)\}$ 
    finsi fait
  retourner VRAI;
fin
```

# Annexe 2-1

## Résolution d'un CSP binaire discret : consistance de chemin

```
procedure PC2(Mp) { %version 2%
  début
  Q ← {(Xi, Xj) ∈ X2 : (i ≤ j) et il existe une contrainte entre Xi et Xj}; entrée_nulle = faux
  tant que Q ≠ ∅ et ¬entrée_nulle faire
    Prendre une paire (Xi, Xj) de variables de Q, et l'en supprimer (Q ← Q \ {(Xi, Xj)}); k = 1
    tant que k ≤ n et ¬(k = i = j) et ¬entrée_nulle faire
      temp = Mp[i, k] ∩ Mp[i, j] ∘ Mp[j, j] ∘ Mp[j, k]; si temp = [0] alors entrée_nulle = true
      si temp ≠ Mp[i, k] alors
        Mp[i, k] = temp; Mp[k, i] = (temp)t
        si i ≤ k alors Q = Q ∪ {(Xi, Xk)} sinon Q = Q ∪ {(Xk, Xi)} finsi finsi
      temp = Mp[k, j] ∩ Mp[k, i] ∘ Mp[i, i] ∘ Mp[i, j]; si temp = [0] alors entrée_nulle = true
      si temp ≠ Mp[k, j] alors
        Mp[k, j] = temp; Mp[j, k] = (temp)t
        si k ≤ j alors Q = Q ∪ {(Xk, Xj)} sinon Q = Q ∪ {(Xj, Xk)} finsi finsi fait fait fin
```

# Annexe 2-1

## Résolution d'un CSP binaire discret : consistance de chemin

fonction Look\_Ahead(M) : booléen

Début

PC2(M)

si inconsistance alors retourner FAUX finsi

si tous les domaines sont singletons alors retourner VRAI

Sinon

choisir une variable disjonctive  $X_i$  de X

pour toute valeur  $V_i$  de  $D(X_i)$  faire

$M' = M$

$M'[i,i][a,b] = \begin{cases} 1 & \text{si } a=b=V_i, \\ 0 & \text{sinon} \end{cases}$

si Look\_Ahead( $M'$ ) alors retourner VRAI finsi

fait

retourner FAUX finsi

fin

Look\_Ahead avec l'algorithme de consistance de chemin PC2 comme procédure de filtrage durant la recherche

# Annexe 2-1

## Résolution d'un CSP binaire discret : consistance de chemin

### Exemple :

Appliquer les algorithmes de consistance de chemin PC1 et PC2 (version 2) au CSP  $P=(X,D,C)$  suivant :

- $X=\{X_1,X_2,X_3\}$
- $D(X_1)=D(X_2)=D(X_3)=\{1,2,3,4,5\}$
- $C=\{c_1 : X_1 < X_2 ;$   
 $c_2 : X_1 < X_3 ;$   
 $c_3 : X_2 < X_3\}$

→ [Voir Annexe 2-2](#)