



**RENAULT**

**Les Applications**

Y. Collette - 64240



**Le problème de séquençement de véhicules RENAULT :  
un problème benchmark revisité d'optimisation sous  
contraintes et une nouvelle aire de jeux pour les  
métaheuristiques ainsi que les méthodes de recherche locales**

**Présentation extraite d'un article de O. Briant, D. Naddef et G. Mounié**

**Y. Collette**



# Plan

- Présentation du problème
- Algorithmes implémentés par les candidats du challenge ROADEF 2005
- Présentation de l'approche recuit simulé



# Le processus de planification de la journée de production dans les usines

Echelle de temps : jour J+5 à mois M+4

Où : usines d'assemblage

Quand : tous les jours

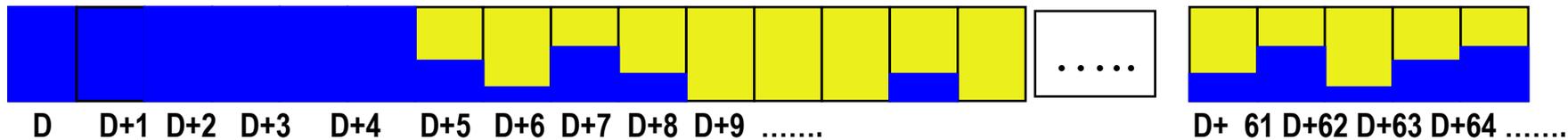


Commandes clients



Commandes pour le stock

## Plan de Production



### Contraintes : Tenir compte des capacités

- des lignes de montage
- des fournisseurs
- des unités de travail

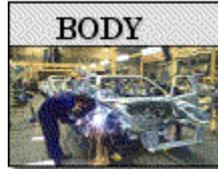
### Objectifs :

- Maximiser le nombre de véh. Accepter dans le plan de production
- Satisfaire les délais de livraison clients
- Assurer la stabilité du plan de prod.



# Plannification d'un jour de production

Séquence de véh. →



↙  
**Lots de caisses**

↕  
**Lots de caisses peintes**

↘  
**Contraintes de Ratio**

▶ **Lots de caisses**

- ✓ Lots de véhicules ayant les mêmes caractéristiques (3-portes, toit ouvrant ...)
- ✓ Minimiser les changements d'outils qui requière des opérations manuelles

▶ **Lors de caisses peintes**

- ✓ Lots de véhicules ayant la même couleur
- ✓ Minimiser la consommation de solvants dus aux changements de couleurs

▶ **Contraintes de Ratio (espacement)**

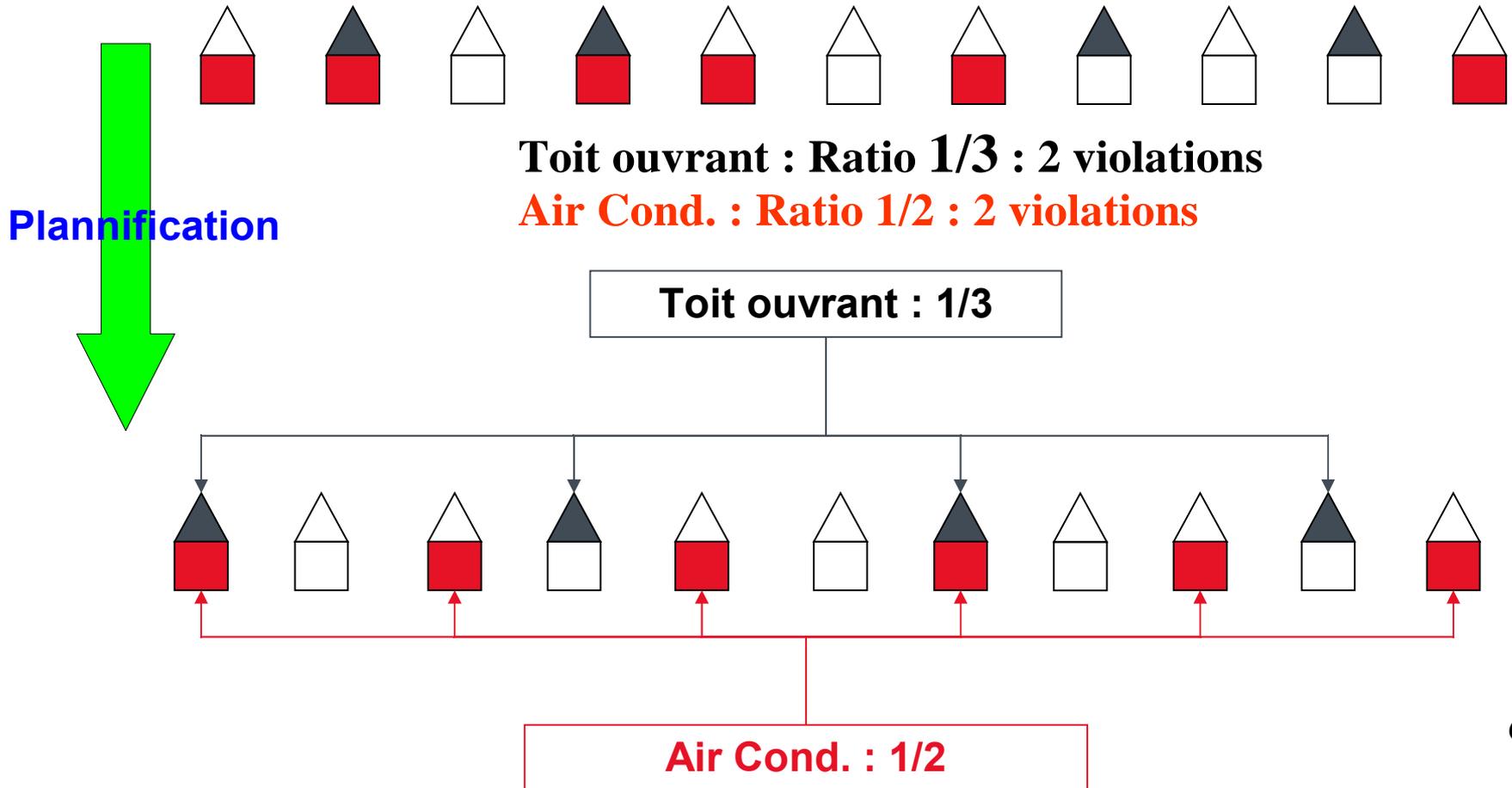
- ✓ Ratio  $N/P$  pour l'option  $i$  : pour chaque séquence consécutives de  $P$  véhicules, au plus  $N$  véhicules requière l'option  $i$
- ✓ Lisser la charge de travail sur la ligne d'assemblage
- ✓ Ne pas surcharger les unités de travail critiques



# Contraintes de Ratio

## Les Écarts de critère

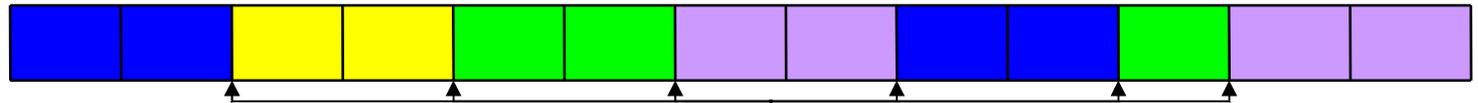
Comment réorganiser une séquence de montage





# Lots de caisses peintes

Comment réorganiser une séquence de peinture



Changement de couleurs : 6

Plannification



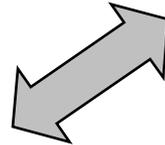
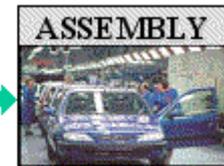
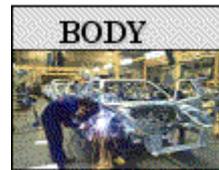
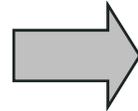
Changement de couleurs : 3



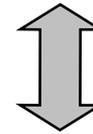


# Le problème soumis au challenge ROADEF

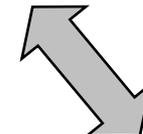
Séquence de véh.



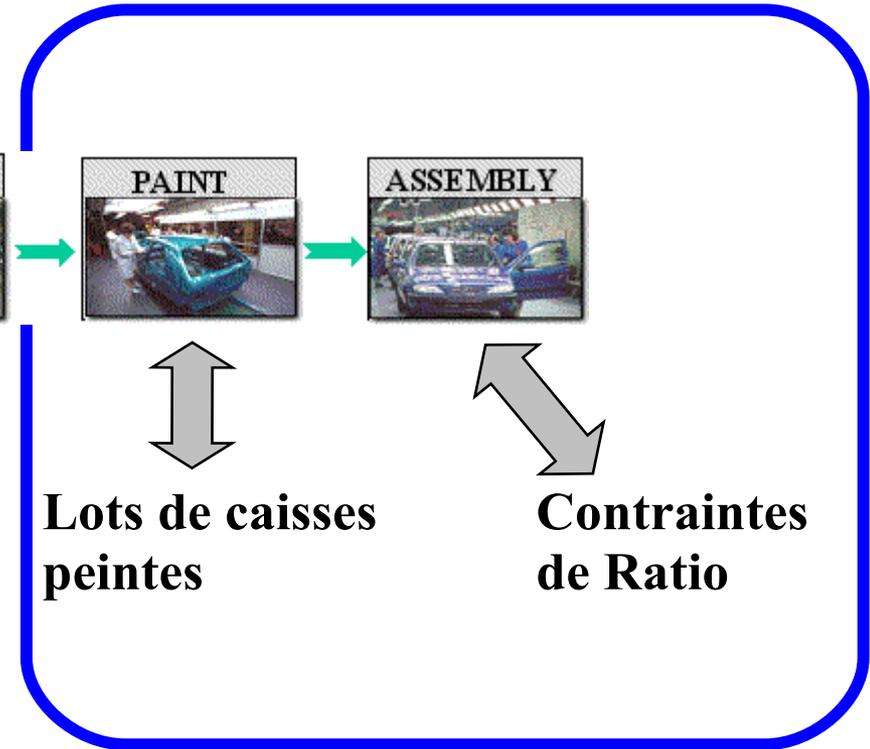
**Lots de caisses**



**Lots de caisses  
peintes**



**Contraintes  
de Ratio**



**Challenge ROADEF 2005**



# Plannification d'un jour de production

## Objectif

Construire une séquence de véh. :

Qui maximise des contraintes « douces » de l'atelier d'assemblage

Qui minimise les coûts de production de l'atelier de peinture

## Contraintes fortes

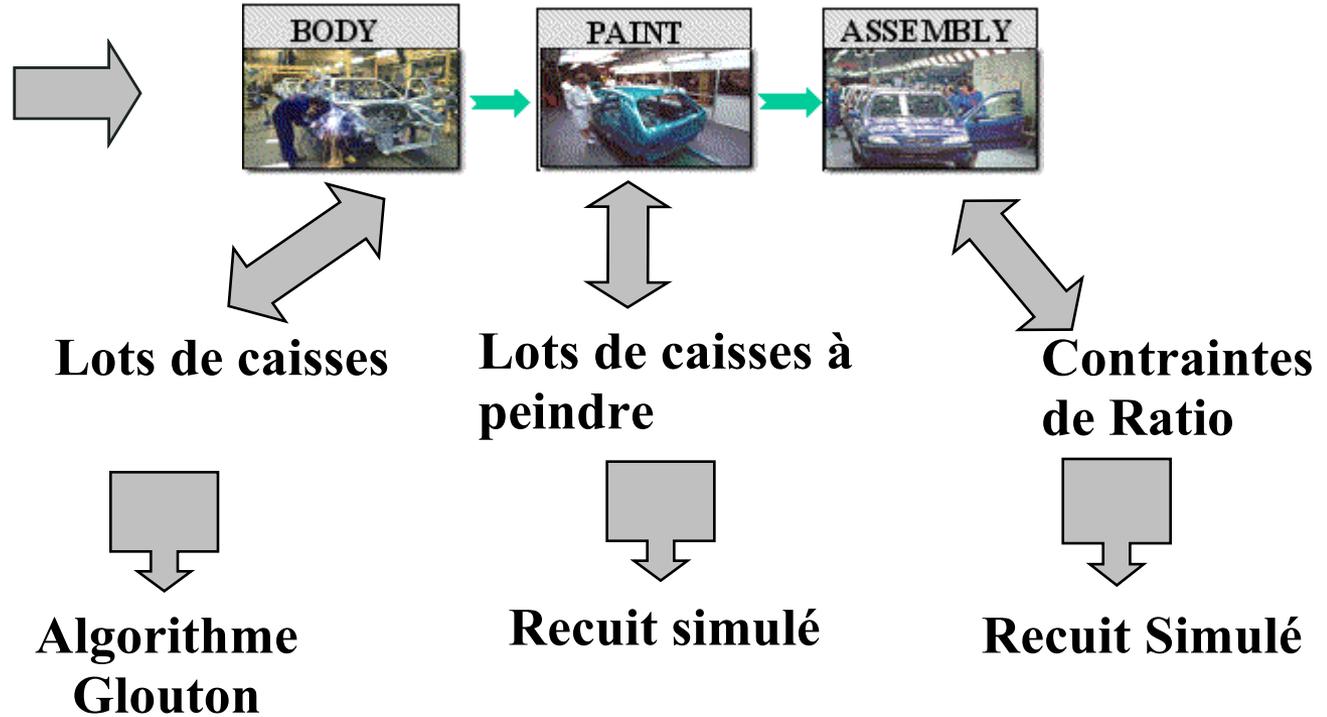
Longueur maximale de la séquence de caisses à peindre

(due aux exigences de nettoyage des installations de peinture)



# Approche RENAULT

Séquence de véh.



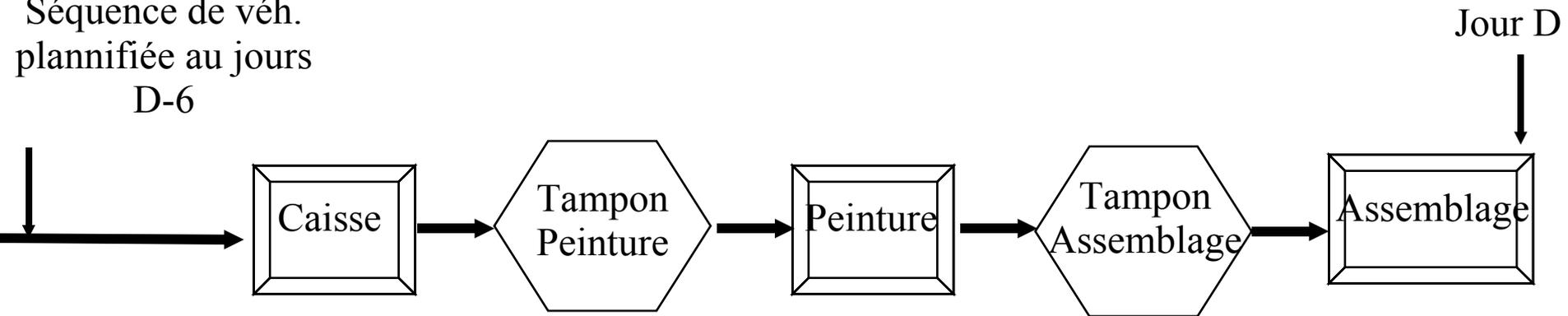
Ce système de séquençement de véhicule est en fonction depuis 1993 dans l'ensemble des usines RENAULT

⇒ Pourquoi un nouvel algorithme de séquençement de véhicules ?



# Le séquençement de véhicules: un nouveau processus critique pour RENAULT

Séquence de véh.  
plannée au jours  
D-6



## ► Stratégie RENAULT : Une compliance stricte vis-à-vis de la séquence de véhicules initiale

- ✓ De façon à assurer une synchronisation avec la livraison des pièces fournisseur (fenêtre de 2 heures émise au jour D-6)
- ✓ Le tampon peinture est limité en taille et le tampon assemblage réordonne la séquence de façon à reproduire la planification initiale
- ✓ La performance de l'algorithme de séquençage est maintenant critique !



# Optimisation Multiobjectif

- Optimisation multicritère lexicographique
  - Pas de compensation entre les objectifs (Contraintes de Ratio et ensuite de caisses à peindre)
  - La priorité entre les fonctions objectifs est donnée par chacune des usines
- Configuration des usines

## Usines à main-d'oeuvre coûteuse (e.g. Europe de l'Ouest)

- Contraintes de ratio à forte priorité (demandes fortes de la part des ateliers de montage)
- Lots de caisses à peindre
- Contraintes de ratio à faible priorité

## Usines à main-d'oeuvre peu coûteuse (e.g. Roumanie/Turquie)

- Lots de caisses à peindre
- Contraintes de ratio à forte priorité
- Contraintes de ratio à faible priorité



# Contraintes de Ratio

- Sens industriel d'une contrainte de ratio  $1/N$  sur l'option  $o_i$ 
  - Pour espacer les véh. ayant l'option  $o_i$  autant que possible à  $N$  positions
  - Quand ce n'est pas possible pour certains véh., les espacer à  $N-1$  positions, puis à  $N-2$  positions etc
  
- Maximiser l'espace entre des véh. qui violent les contraintes de ratio
  - ⇒ Compter les violations en utilisant une fenêtre glissante
  
- Affecter des pénalités lourdes aux véhicules qui sont trop proches

Exemple Contrainte de Ratio =  $1/5$

Séquences (a)  $\_ X \_ \_ \_ X \_$  (b)  $\_ \_ X \_ \_ X \_$

⇒ La séquence (a) est meilleure que la séquence (b)

(f)  $\_ X \_ \_ \_ X \_ : 1$  violation dans la fenêtre  $\_ [X \_ \_ \_ X] \_$

(g)  $\_ \_ X \_ \_ X \_ : 1$  violation dans la fenêtre  $\_ [ \_ X \_ \_ X ] \_$   
 + 1 violation dans la fenêtre  $\_ \_ [X \_ \_ X \_ ]$

⇒ nb. de séquence violant la contrainte pour (a) < nb. de séquences violant la contrainte pour (b)



# Fonction objectif pour les contraintes de Ratio

## Notations

$n_{car}$  : le nombre total de véhicules

$n_{opt}$  : le nombre total d'options

Contraintes de Ratio  $i$  :  $p_i / q_i$

$c_{ki}$  : Paramètre booléen indiquant si le véh.  $k$  requière l'option  $i$

$V_{ij}$  : Nombre de violations de contraintes de ratio  $i$  dans la séquence commençant à la position  $j$  et de longueur  $q_i$

$Y_{kj}$  : Paramètre booléen indiquant si le véh. est positionné en  $j$

$$\text{Min } F = \sum_{i=1}^{n_{opt}} \sum_{j=1}^{n_{car}-q_i+1} V_{ij}$$

$$\text{où } V_{ij} = 0 \quad \text{si} \quad \sum_{s=j}^{j+q_i-1} \sum_{k=1}^{n_{car}} c_{ki} * Y_{ks} \leq p_i$$

$$= \sum_{s=j}^{j+q_i-1} \sum_{k=1}^{n_{car}} c_{ki} * Y_{ks} - p_i \quad \text{autrement (1 dans le modèle classique)}$$

Minimiser les violations de contraintes sur l'ensemble des véh. et l'ensemble des options

S'il n'y a pas de violation de contraintes, on met 0

S'il y a violation de contraintes, on met le dépassement de la contrainte

Ancienne méthode RENAULT de décompte

⇒ La fonction objectif est **le nombre de véhicules qui violent la contrainte de ratio** (et non pas le nombre de contraintes de ratio violées comme dans le modèle classique !)



# Contraintes de Ratio

## Exemple de calcul

Exemple Contrainte de Ratio = 1/5

Séquences (a) XX X\_ \_ et (b) \_ X \_ X \_ X \_

⇒ séquence (b) est meilleure que la séquence (a)

- (a)
- \_\_ X X \_ X \_ \_
  - [\_\_ XX] X \_ \_ : 1 violation RENAULT (1 violation dans le cas classique)
  - [\_ XX \_] X \_ \_ : 1 violation RENAULT (1 violation dans le cas classique)
  - \_ [XX \_ X] \_ \_ : 2 violations RENAULT (1 violation dans le cas classique)
  - \_\_ [XX \_ X \_] \_ : 2 violation sRENAULT (1 violation dans le cas classique)
  - \_\_ X [X \_ X \_ \_] : 1 violation RENAULT (1 violation dans le cas classique)

total : **7 violations RENAULT** (5 violations dans le cas classique)

- (b)
- \_ X \_ X \_ X \_
  - [\_ X \_ X] X \_ : 1 violation RENAULT (1 violation dans le cas classique)
  - [\_ X \_ X \_] X \_ : 1 violation RENAULT (1 violation dans le cas classique)
  - \_ [X \_ X \_ X] \_ \_ : 2 violations RENAULT (1 violation dans le cas classique)
  - \_ X [X \_ X \_] \_ : 1 violation RENAULT (1 violation dans le cas classique)
  - \_ X \_ [X \_ X \_ \_] : 1 violation RENAULT (1 violation dans le cas classique)

total : **6 violations RENAULT** (5 violations dans le cas classique)

⇒ nb. de séquences de violations (b) < nb. de séquences de violations (a)



# Environnement industriel

- Limite temporelle : **10 minutes** de temps CPU sur un Pentium-IV
- Problème de grande tailles

Usines RENAULT	Véh.	options & couleurs	ratio avec taille max block	option/configurations couleurs
022	485	11 options / 15 couleurs	10/15	58
064-2	335	6 options / 12 couleurs	1/100	37
064-1	875	9 options / 14 couleurs	1/100	133
039	954	5 options / 14 couleurs	1/297	113
048	600	17 options / 12 couleurs	1/274	229
024	1260	13 options / 13 couleurs	1/20	237
025	485	22 options / 24 couleurs	1/76	287

CSP lib	Véh.	options & couleurs	ratio avec taille max block	option/configurations couleurs
Instancesclassiques	100-200	5 options	2/5	17-30
Nouvelles instances	200-400	5 options	2/5	19-26



# Differences avec le problème classique de séquençement de véhicules

- ✓ Optimisation multiobjectif
- ✓ Prise en compte des objectifs de l'atelier de peinture
- ✓ Fonction objectif modifiée pour les contraintes de Ratio
- ✓ Prise en compte des séquences à jour D-1
- ✓ Problèmes de grandes tailles
- ✓ Un problème d'optimisation et non pas un problème de satisfaction de contraintes
  - ⇒ *Même s'il peut subsister des violations de contraintes de ratio, la séquence de véhicules doit être produite !*



- Présentation du problème
- Algorithmes implémentés par les candidats du challenge ROADEF 2005
- Présentation de l'approche recuit simulé



# Méthodes implémentées par les 25 équipes candidates

Méthodes	Nombre d'équipes
Méthode Glouton pour construire la solution initiale	Toutes sauf 1
Recuit simulé	5
Recherche Tabou	7
Voisinage variable / Recherche à large voisinage / Recherche locale itérée	13
Algorithme génétique	1
Colonies de fourmis	1
Programmation linéaire en nombres entiers	1



# Méthodes des 12 meilleures équipes

Place	Méthodes	Equipe
1	Algorithme Glouton + Recherche locale	Estellon, Gardi et Nouioua
2	Algorithme Glouton + Recherche à voisinage variable + Recherche locale itérée	Aloise, Noronha, Rocha, Ribeiro et Urrutia
3	Algorithme Glouton + Recuit simulé	Briant, Naddef et Mounié
4	Algorithme Glouton + Recuit simulé	Bloemen
5	Algorithme Glouton + Recuit simulé	Kuipers
6	Algorithme glouton + Recherche à voisinage variable + Recherche Tabou	Gavranovic
7	Recherche Tabou	Cordeau, Laporte et Pasin
8	Algorithme Glouton + Recherche à voisinage variable + Recuit simulé	Riesler, Chiarandini, Paquete, Schiavinotto et Stützle
9	Algorithme Glouton + Recherche Tabou	Craciunas, Gendreau et Potvin
10	Algorithme Glouton + Recherche locale	Pawlak, Rucinski, Piechowiak, Plaza
11	Algorithme Glouton + Colonie de fourmis + Rech. Tabou + Recherche à voisinage variable	Gravel, Gagné, Price, Krajecki et Jaillet
12	Algorithme Glouton + Recherche locale	Benoist



- Présentation du problème
- Algorithmes implémentés par les candidats du challenge ROADEF 2005
- **Présentation de l'approche recuit simulé**



# Méthode d'optimisation pour le séquençement des véhicules

Algorithme en 2 phases:

Phase 1: Algorithme glouton (pour les couleurs et les ratios prioritaires)

Phase 2: Recuit simulé multiobjectif

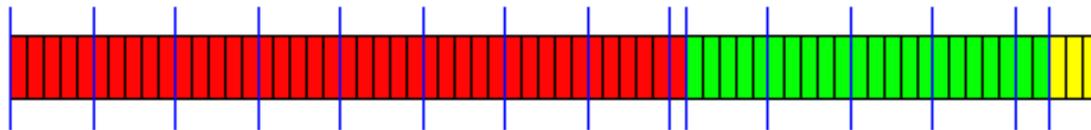


# Algorithme Glouton pour les couleurs

Trie des voitures par couleurs:



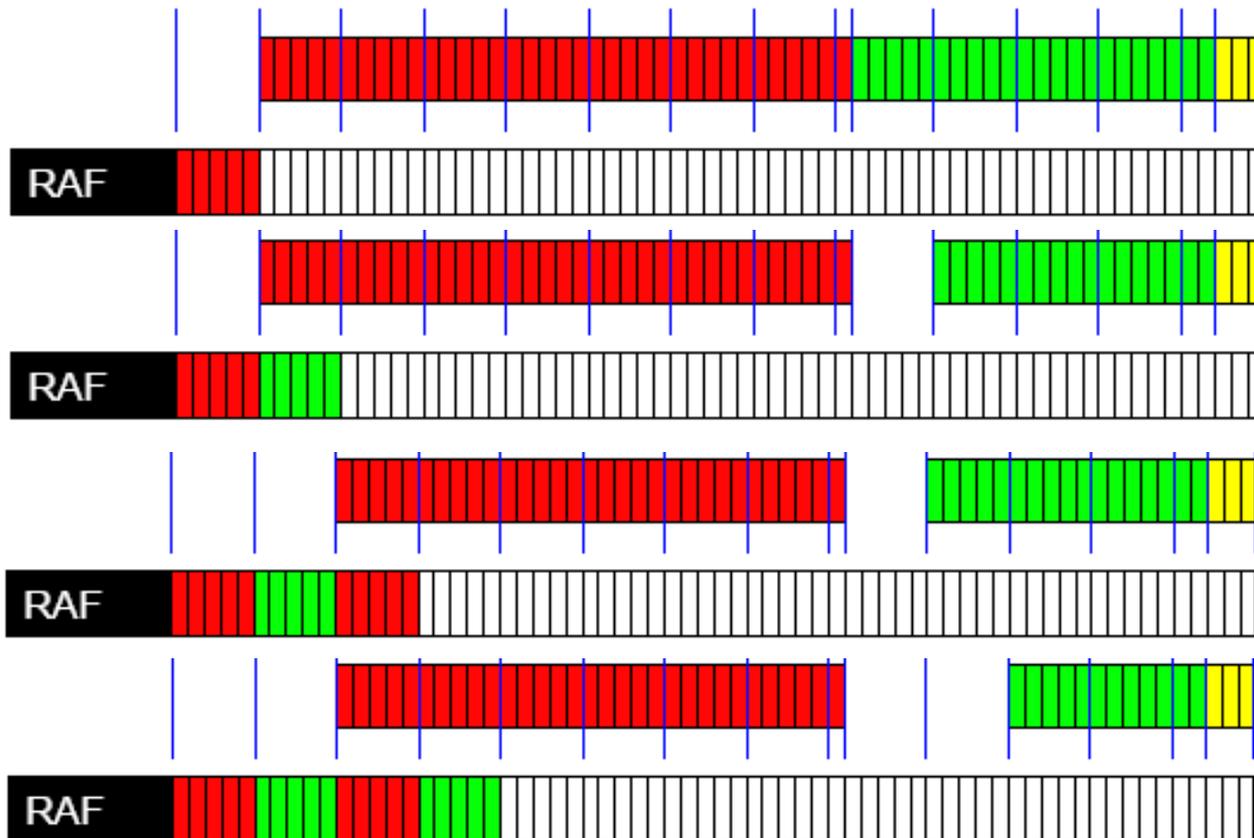
Sélection de blocs d'une même couleur de la plus grande taille possible:





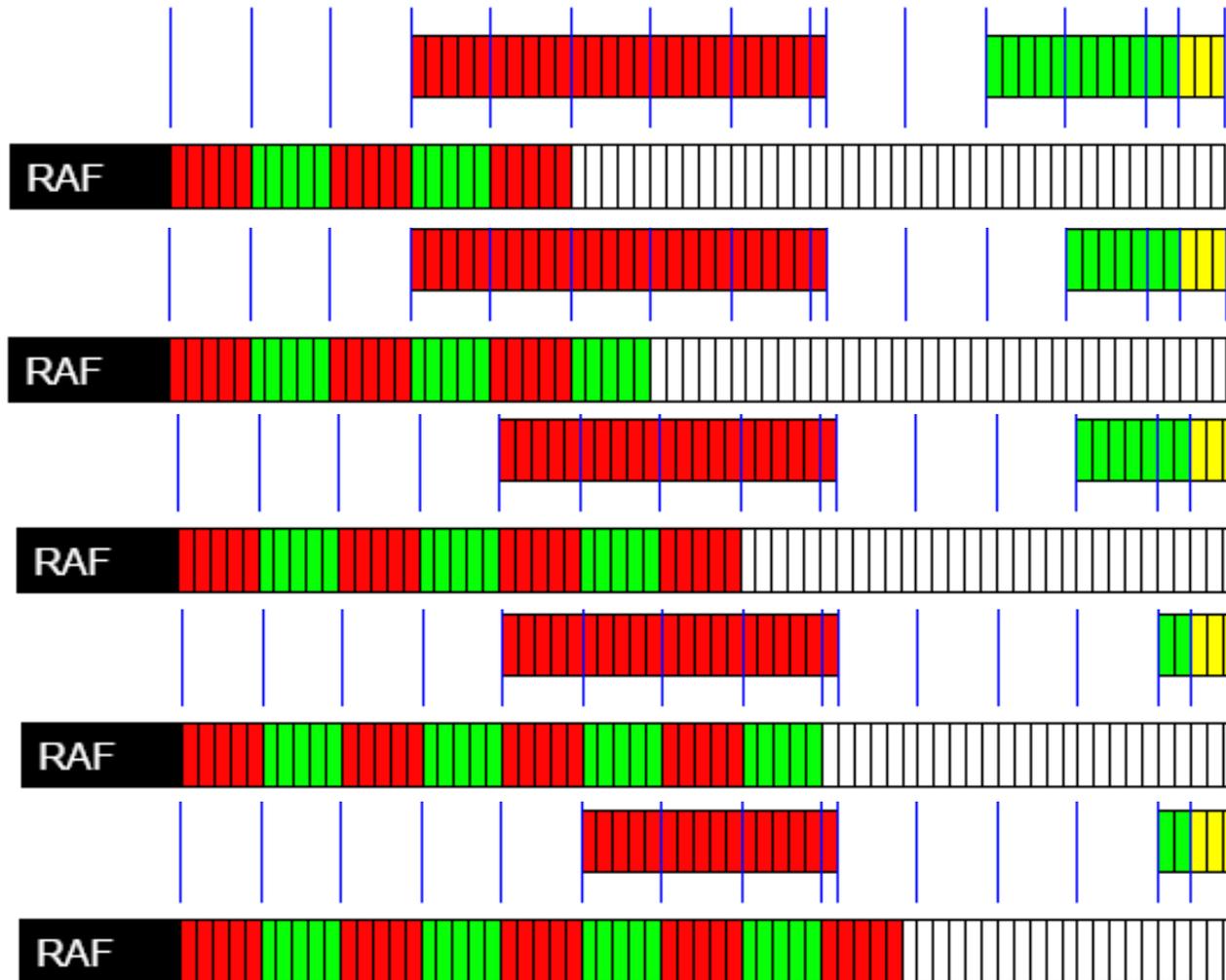
# Algorithme Glouton pour les couleurs

Allocation du bloc de couleur qui reste le plus et qui est différent du dernier bloc





# Algorithme Glouton pour les couleurs

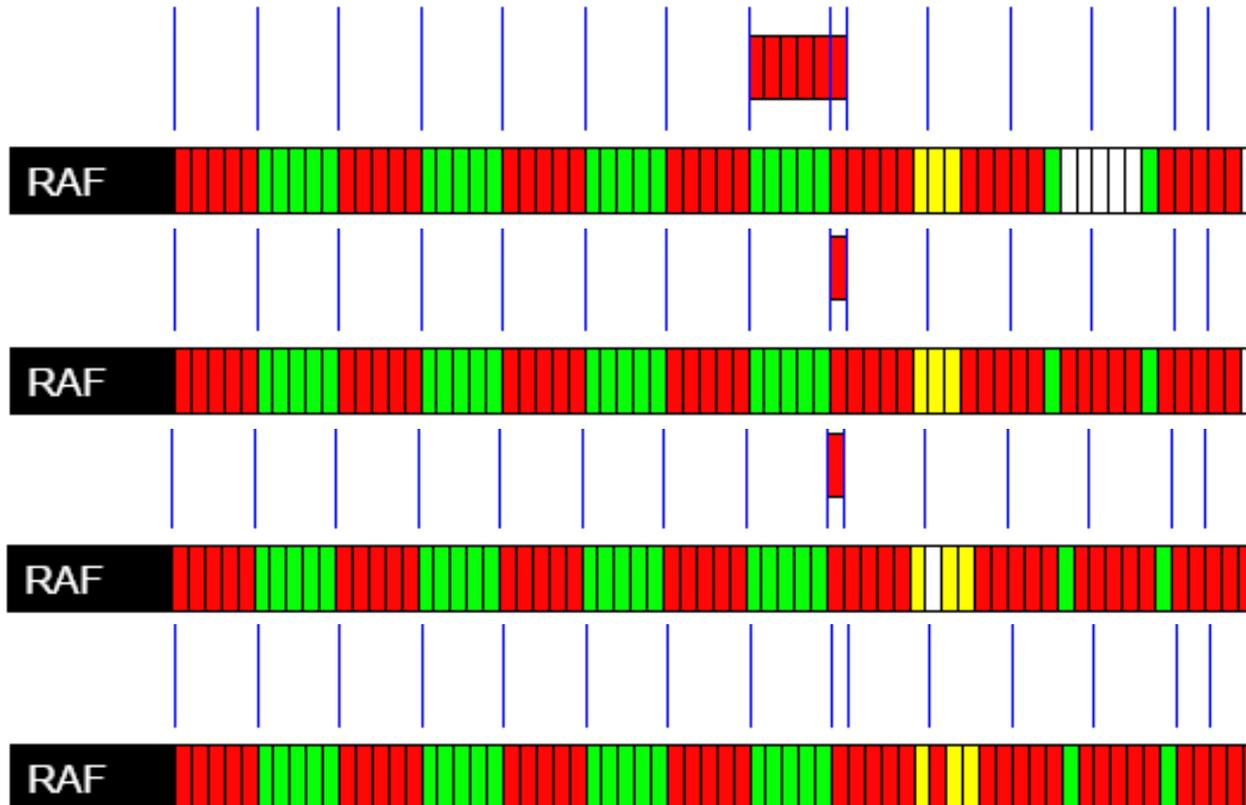






# Algorithme Glouton pour les couleurs

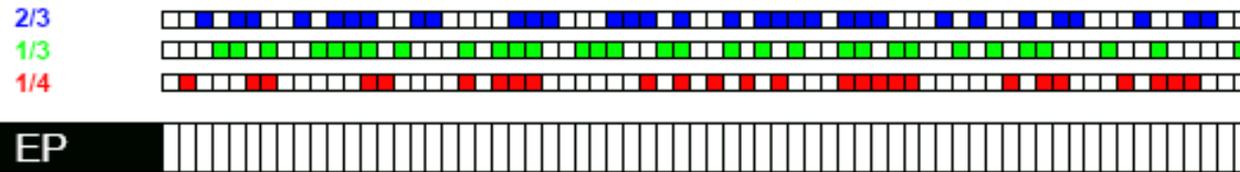
S'il reste un bloc d'une même couleur, découper un bloc d'une autre couleur



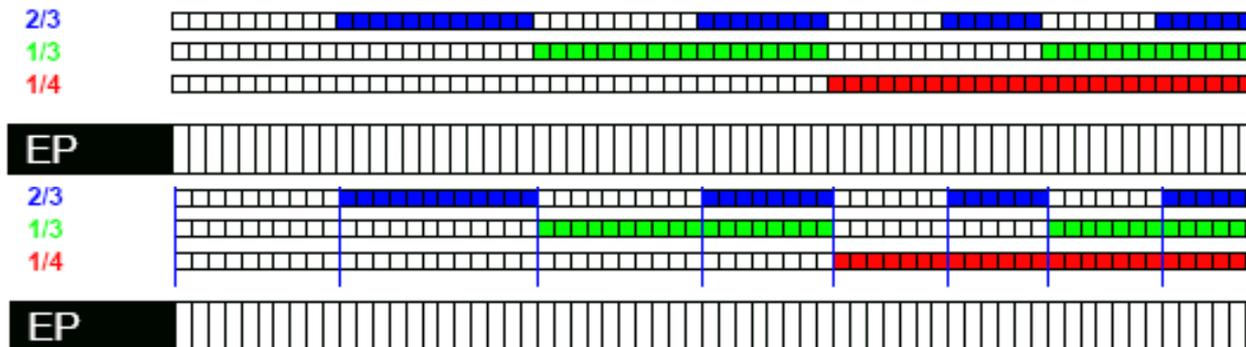


# Algorithme Glouton pour les ratios

Algorithme glouton avec priorisation dynamique



Les véhicules sont classés par famille. Les véhicules concernés par le même ratio font parti de la même famille



EP: ratio prioritaire

ENP: ratio moins prioritaire

RAF: nombre de purges



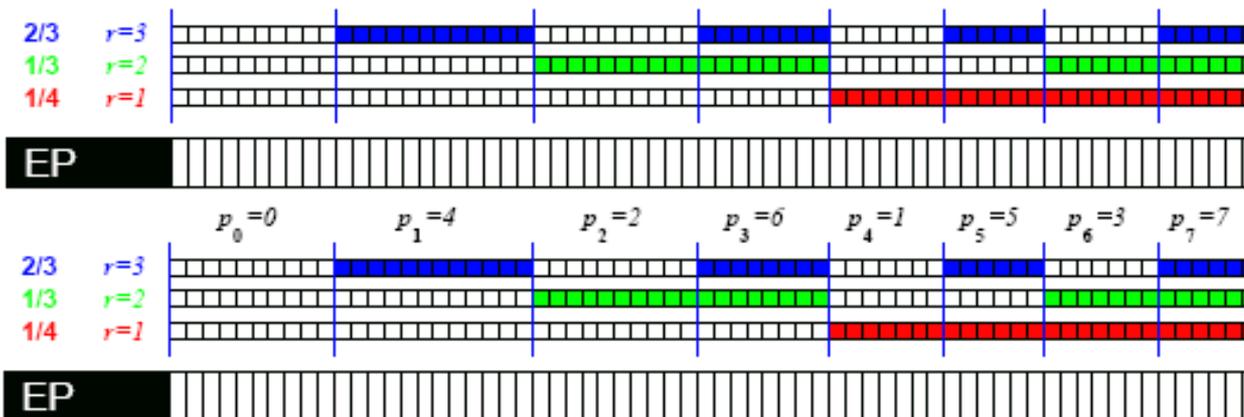
# Algorithme Glouton pour les ratios

A intervalles réguliers, les ratios sont classés en ordre croissants de distance par rapport à la borne de la contrainte.

Soit  $r_j$  le rang du ratio  $j$ . Chaque famille  $k$  de véhicules se voit affecté une priorité

$$p_k = \sum \left\{ 2^{r_j - 1} : \text{la famille } k \text{ contient le ratio } j \right\}$$

On planifie ensuite les véhicules les uns après les autres.



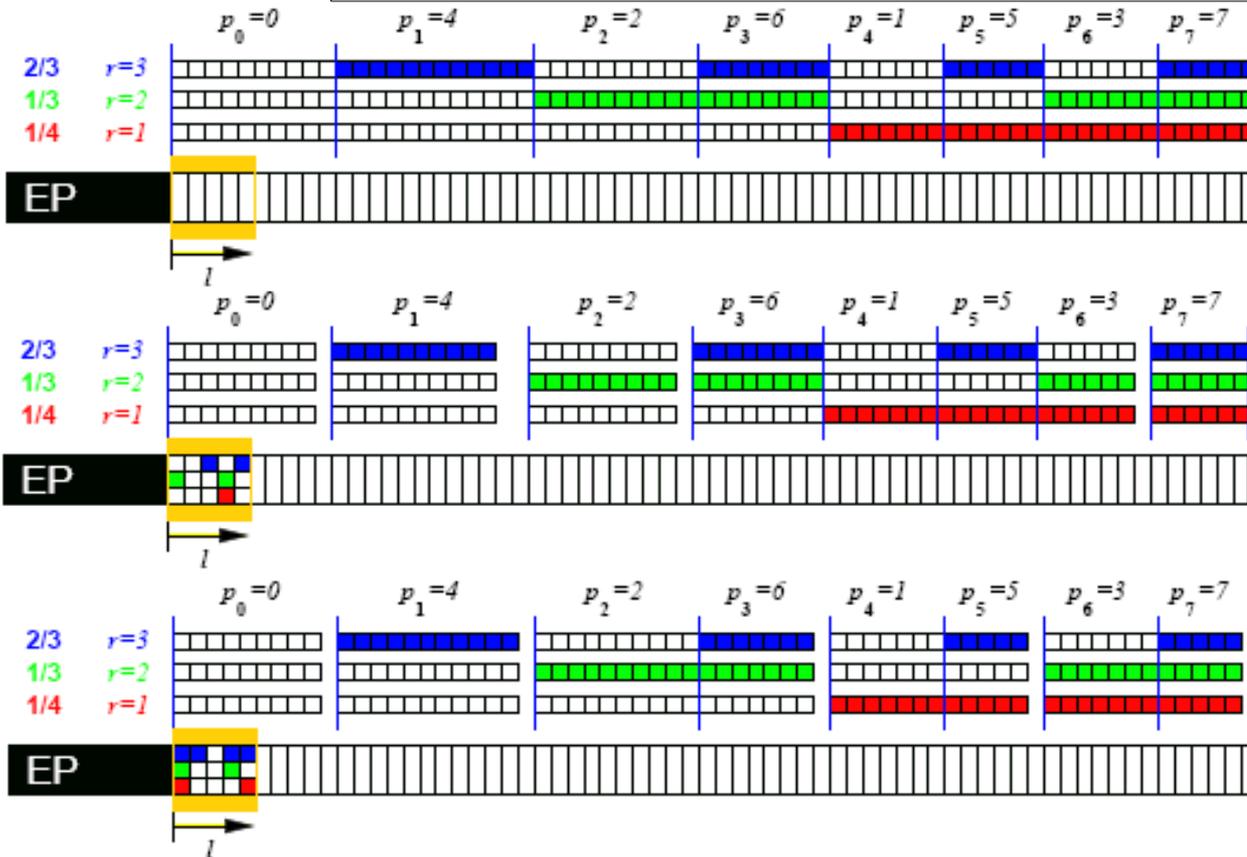
Une famille est d'autant plus prioritaire qu'elle est composée de contraintes de ratio de rang élevé



# Algorithme Glouton pour les ratios

Exploration de toutes les combinaisons d'affectation  $\langle k_1, k_2, \dots, k_l \rangle$  sans violation dans les prochaines  $l$  positions, et stocker les affectations qui maximisent:

$$f(\langle k_1, k_2, \dots, k_l \rangle) = \sum_{i=1}^l \alpha^i p_{k_i} \text{ avec } \alpha < 1$$



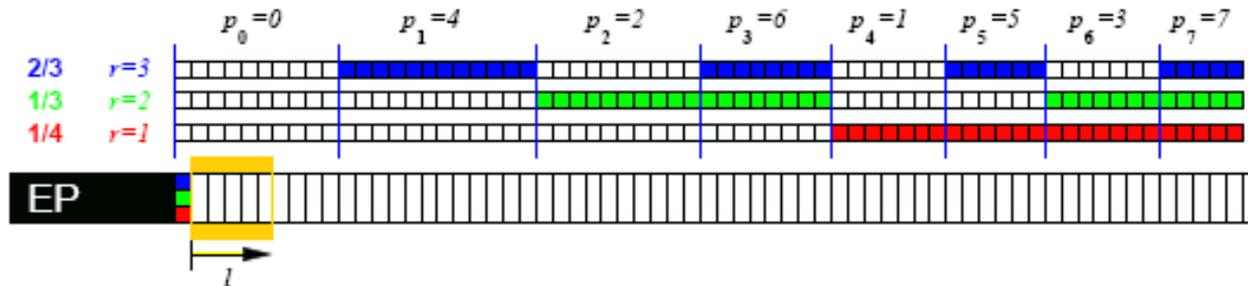
Cette fonction objectif permet de caser les familles non-prioritaires au début de la séquence

De cette façon, on construit « facilement » des séquences de longueur  $l$  qui ne violent pas les contraintes de ratio



# Algorithme Glouton pour les ratios

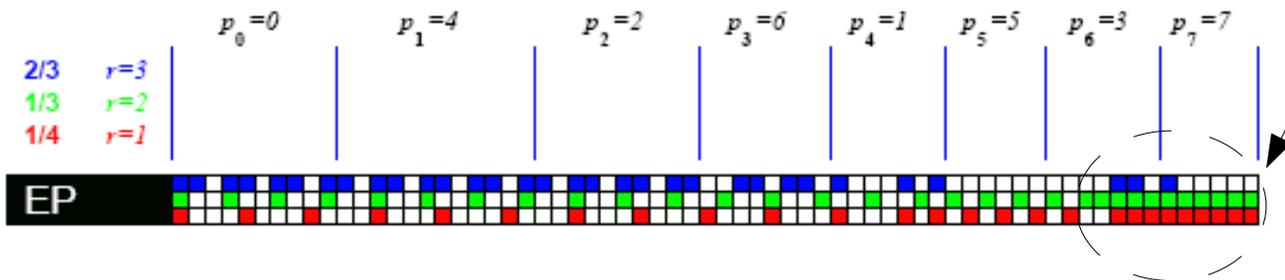
On affecte les premiers  $k_1$  véhicules de cette séquence



La valeur de  $l$  dépend du nombre de ratios. Une valeur entre 4 et 7 semble bien marcher. Si on ne peut pas éviter les violations, on sélectionne l'affectation qui minimise cette violation.

Cette procédure prédictive tend à repousser un maximum la violation des contraintes. On se retrouve alors avec les violations à la fin.

Il semble que cette structure de solution soit efficacement traitée par les méthodes d'amélioration comme le recuit simulé.



On se retrouve avec une séquence bonne au début mais dont la structure ne vérifie pas les contraintes de ratio à la fin !



# Recuit simulé multiobjectif

Procédure d'optimisation:

- Optimisation de la fonction objectif  $k=2$ , puis 3
- Aucune détérioration de la fonction objectif  $k_0 < k$  n'est autorisée
- On passe le même temps pour l'optimisation de chacun des objectifs

Réglage du recuit simulé:

- La température initiale est choisie de façon à ce que la probabilité d'accepter un accroissement d'objectif de 1% soit moins que 0.5%
- Le coefficient de réduction de la température est choisi de façon à ce qu'après  $2/3$  du temps d'affectation pour l'objectif  $k$ , le recuit est gelé



# Recuit simulé multiobjectif

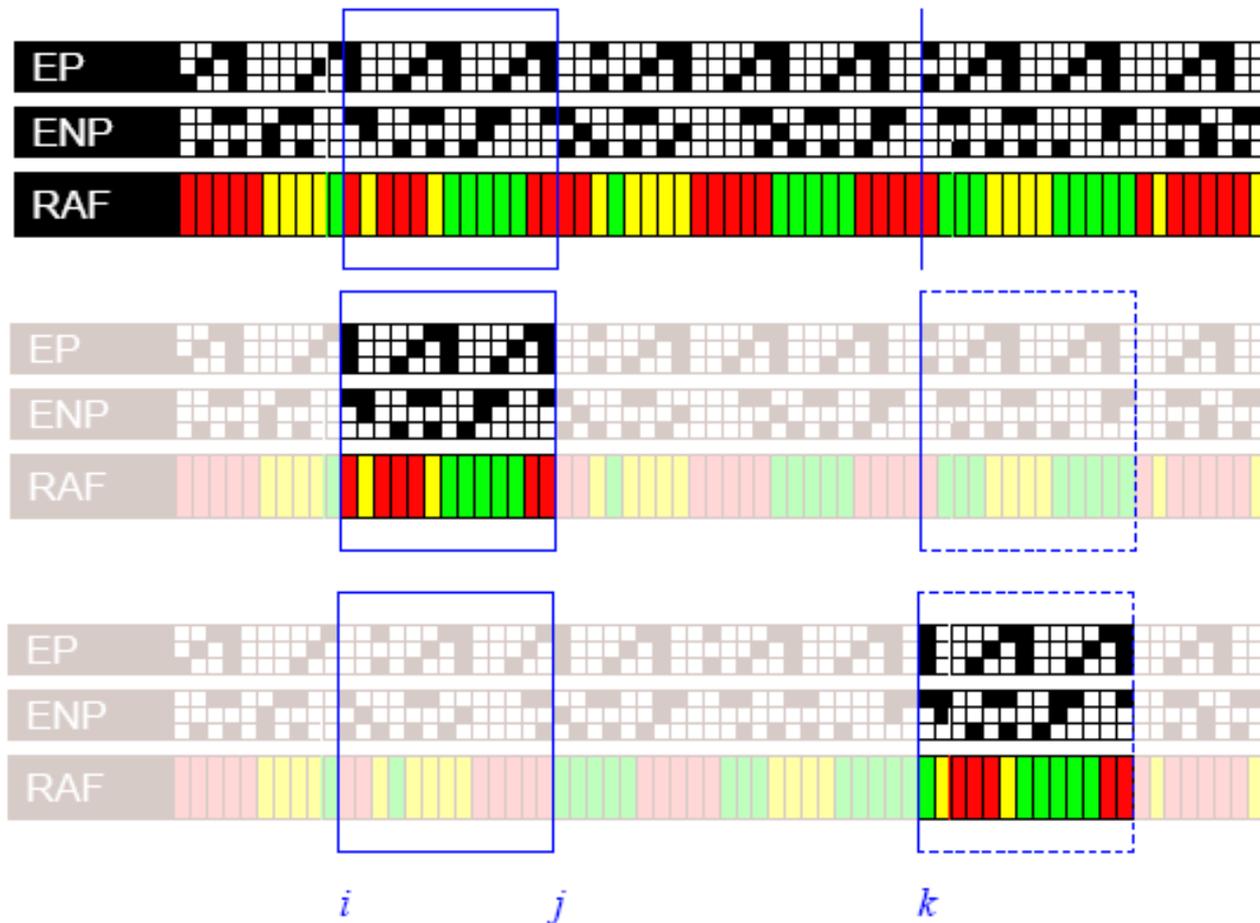
Comment construire un opérateur de voisinage pour le recuit simulé ?





# Recuit simulé multiobjectif

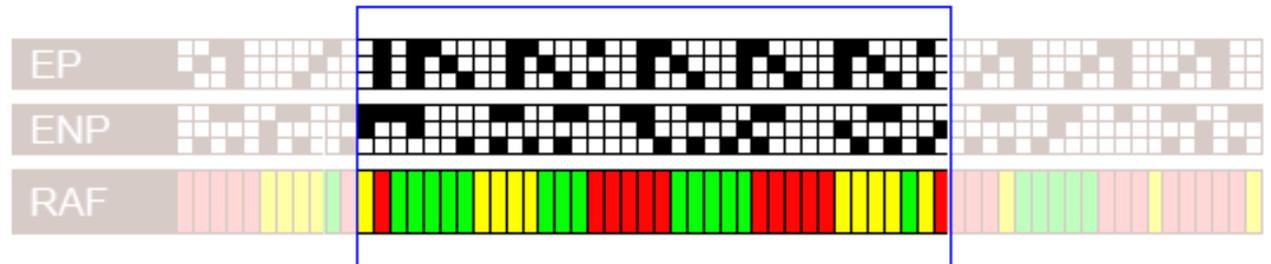
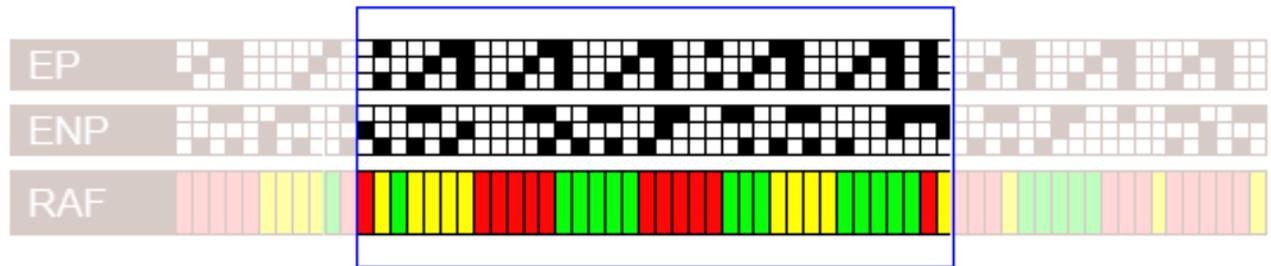
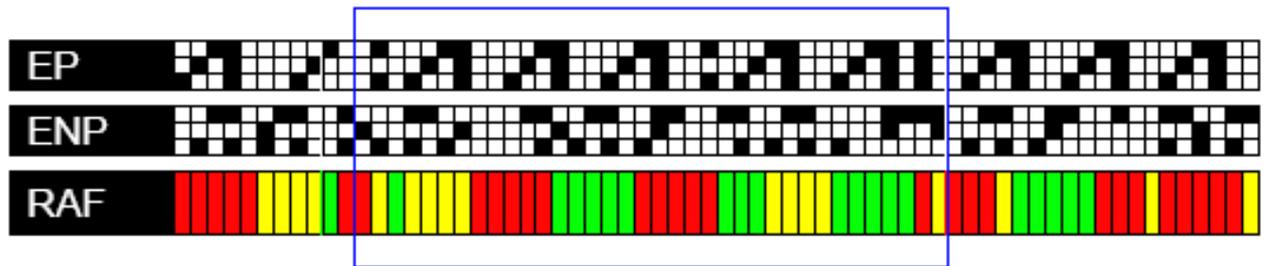
DECBLOC: décalage d'un bloc  $[i,j]$  en position  $k$





# Recuit simulé multiobjectif

INVBLOC: inversion d'un bloc [i,j]



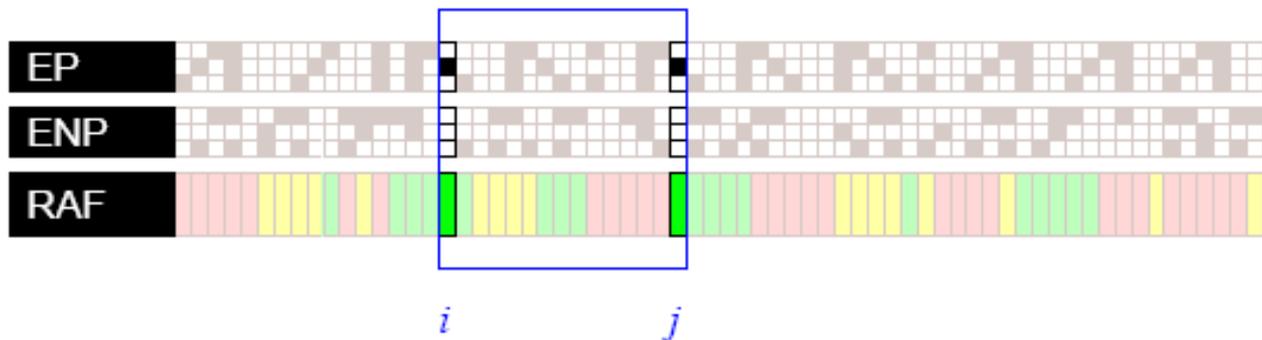
*i*

*j*

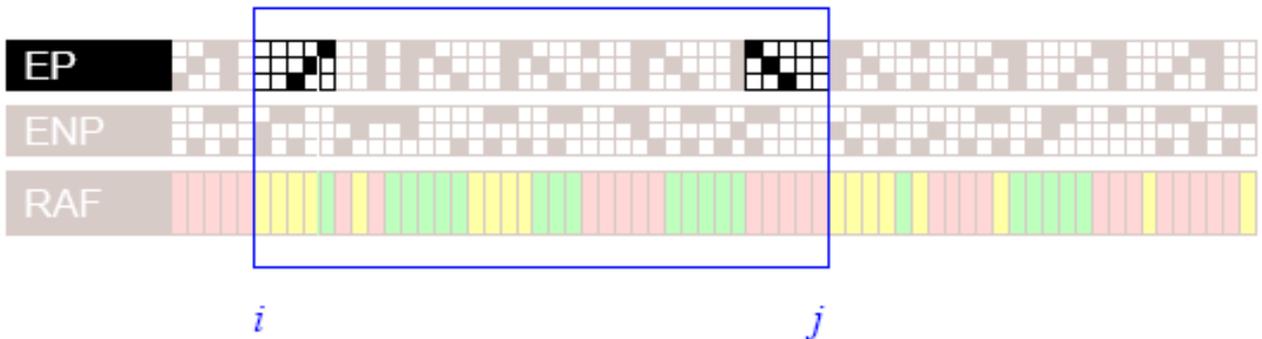


# Recuit simulé multiobjectif

INV\_TYPE:  $i$  et  $j$  du même type (EP, ENP, RAF)



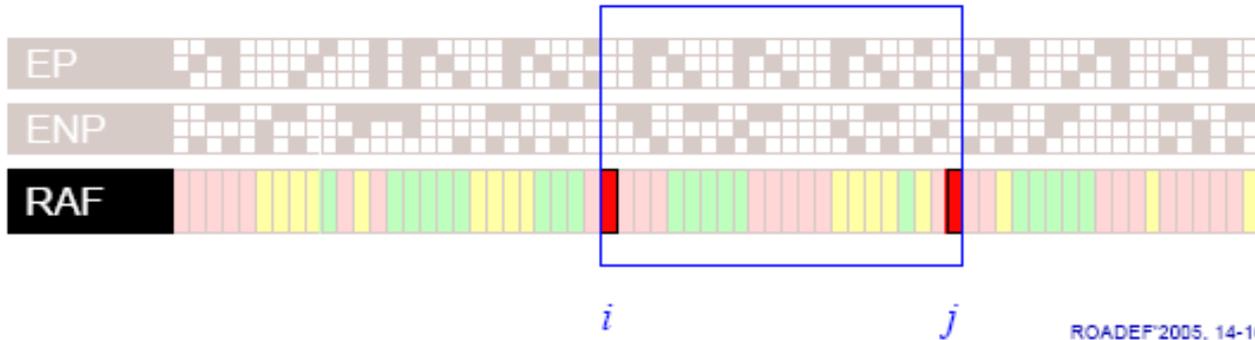
INV\_EP:  $i$  et  $j$  du même type EP



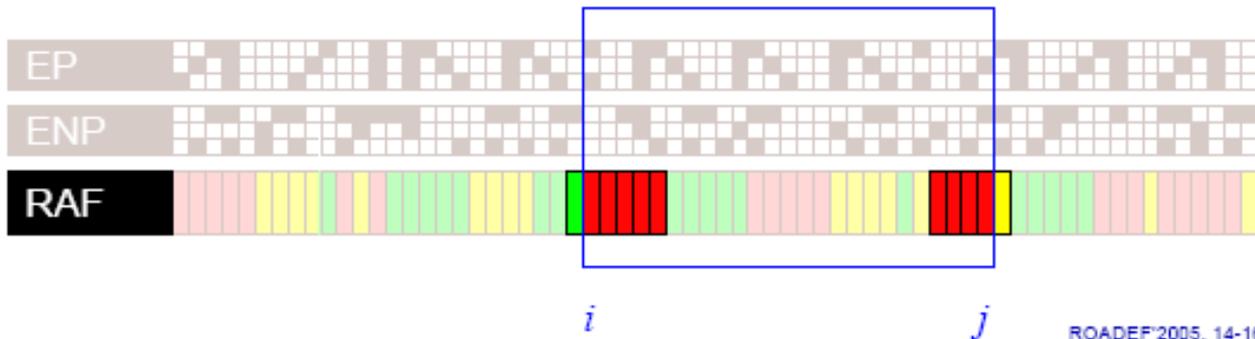


# Recuit simulé multiobjectif

INV\_RAF:  $i$  et  $j$  de la même couleur



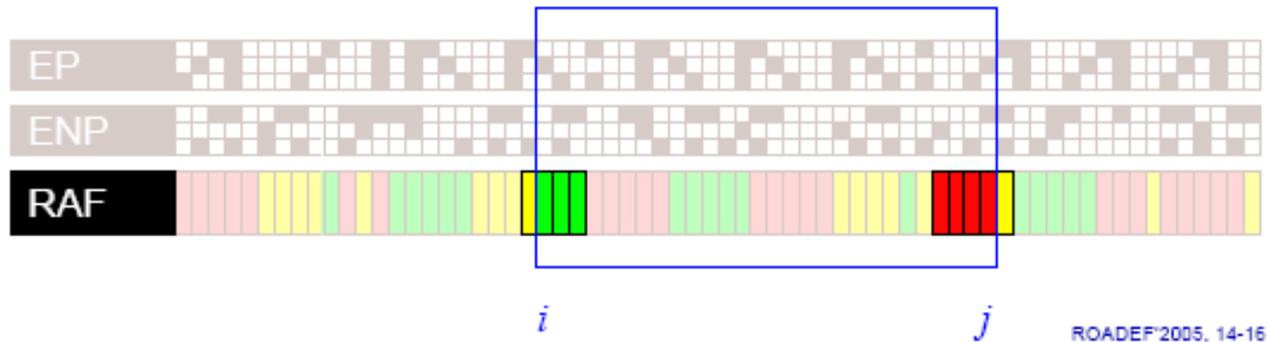
INV\_RAF\_BL1:  $i$  et  $j$  sont les extrémités ayant la même couleur, de blocs de la même couleur





# Recuit simulé multiobjectif

INV\_RAF\_BL2:  $i$  et  $j$  sont les extrémités ayant des couleurs différentes de blocs de la même couleur





# Recuit simulé multiobjectif

Comment choisir le meilleur opérateur de voisinage pour le recuit simulé ?



# Recuit simulé multiobjectif

Test de 200 ECHBIN après chaque mouvement

Au début de la phase 2, tous les mouvements (à part ECHBIN) ont la même priorité ( $\text{Pr}[i] = 1/7$ )

Chaque seconde, après 20 mouvements efficaces, mise à jour des probabilités:

$$P[i] = \beta P[i] + (1 - \beta) P'[i] \text{ avec } P'[i] = \frac{\text{nbSuccess}[i] / P[i]}{\sum_j \text{nbSuccess}[j] / P[j]}$$

et  $\beta$  est mis à jour de la façon suivante:  $\beta = \beta \cdot 0.992$

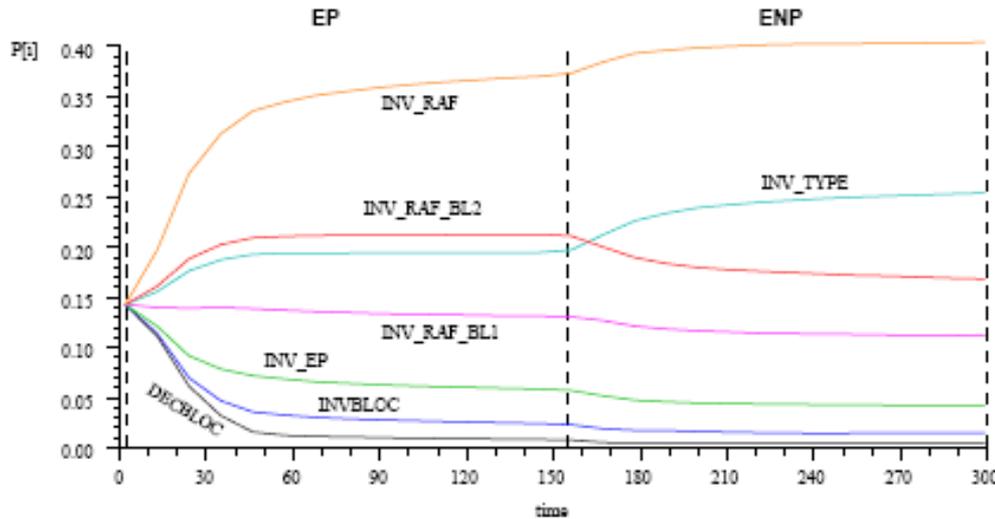
Au début de l'optimisation de l'objectif  $k$ ,  $\beta = 0.95$  et  $\text{nbSuccess}[i]=0$  pour chaque mouvement.

La probabilité dépend de:

- L'objectif déjà optimisé  $k' < k$ , et de l'objectif  $k$
- La complexité de l'instance elle-même

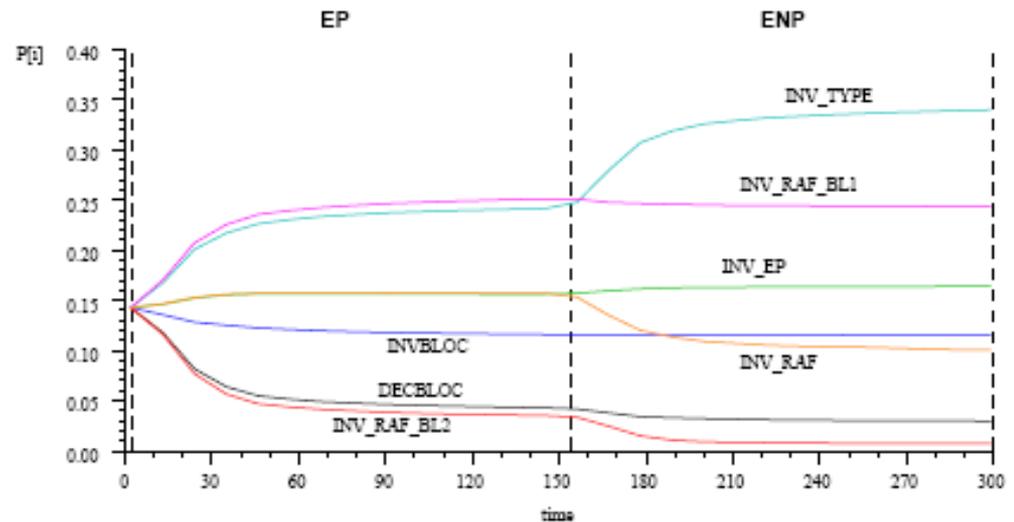


# Evolution du type de voisinage utilisé



Sur l'instance n°1

Sur l'instance n°2



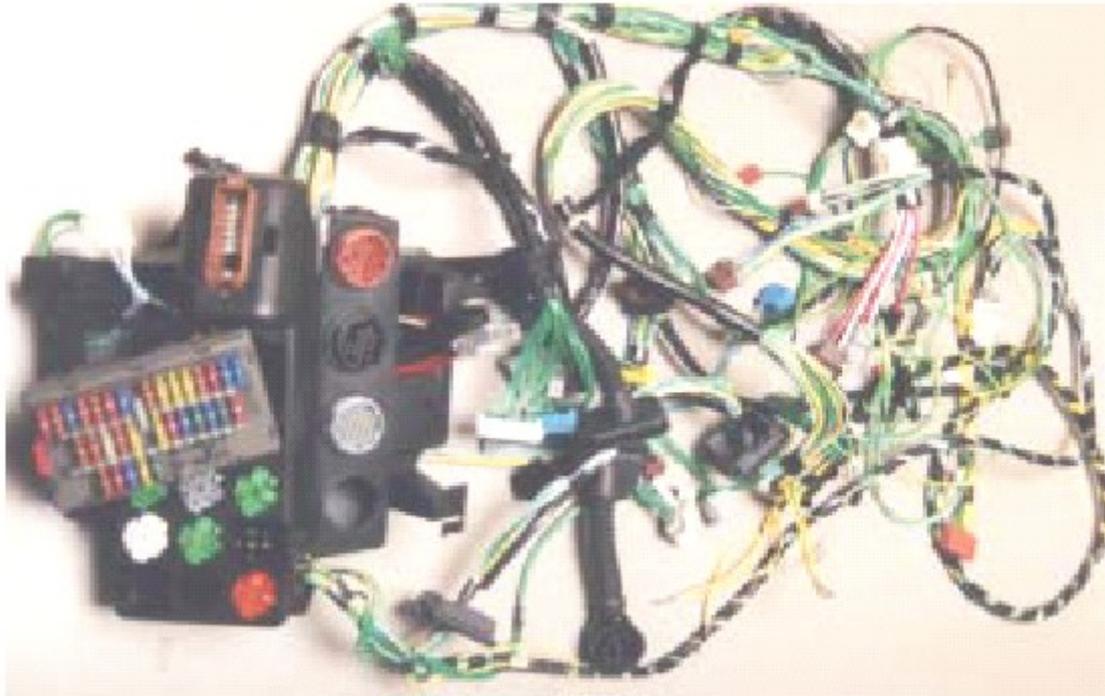


# Conclusions

- Déploiement de la méthode de recherche locale issue du challenge ROADEF dans les usines RENAULT le 15 Nov 2005
- Nouvelles contraintes à explorer :
  - Contrainte sur la position d'un véhicule (*p.ex. Les véhicules avec la nouvelle option i doivent être produits le matin*)
  - Succession de couleurs interdites (*p.ex. Un véhicule noir ne doit pas suivre ou précéder un véhicule blanc*)
  - Plusieurs lignes de peinture pour la même ligne de montage (*nouvelle méthode de calcul pour le changement de peinture*)

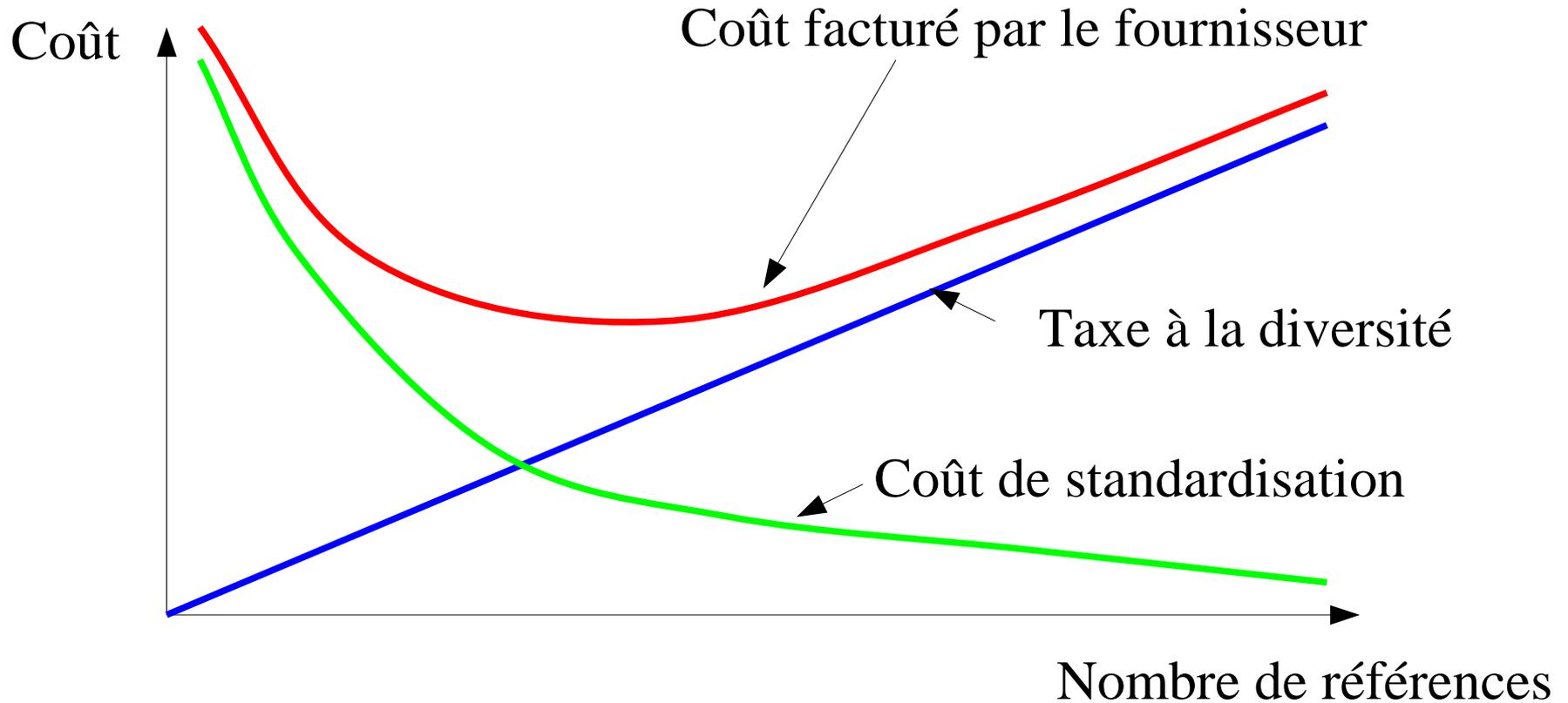


# TOSCA



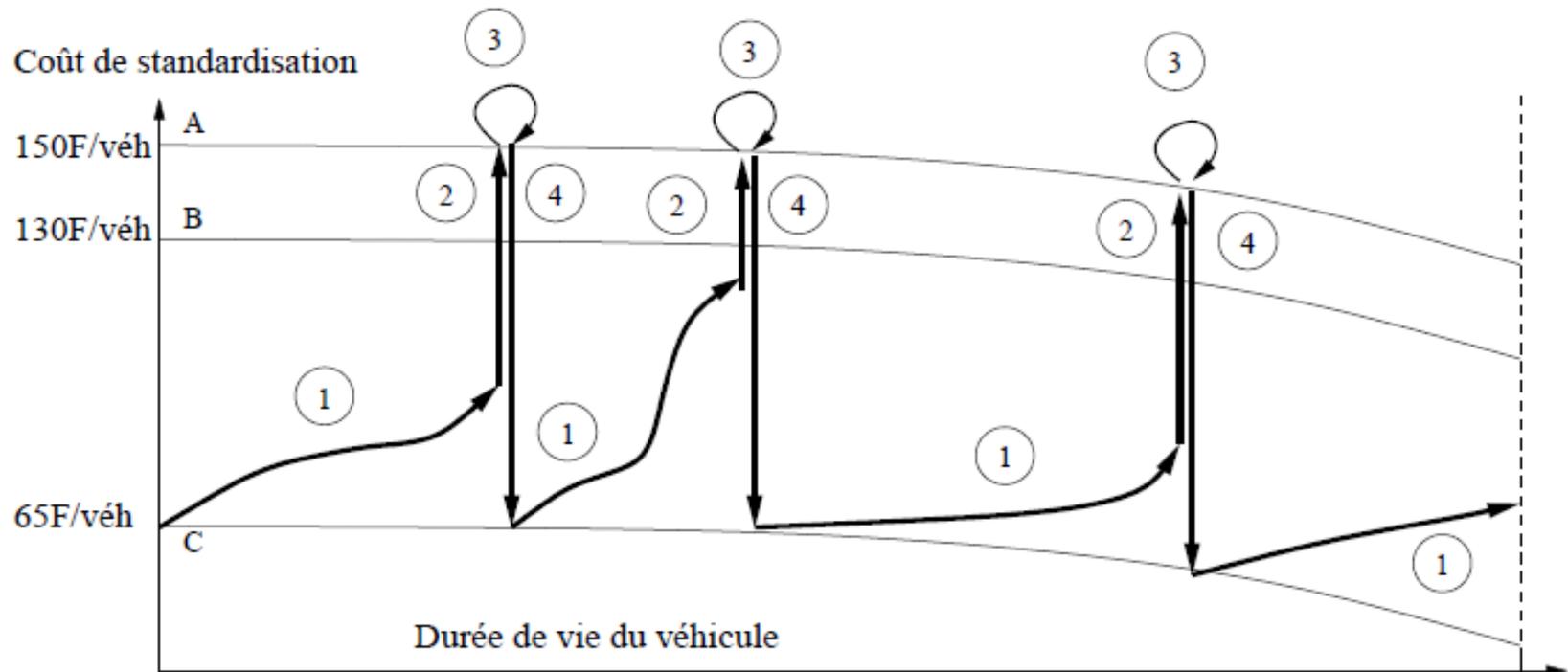


# TOSCA: Le coût de production des câblages





# TOSCA: Le problème





# Plan de la présentation

- 1 Problématique TOSCA
- 2 Méthode de résolution actuelle
- 3 Le voisinage pour le recuit simulé
- 4 Perspectives



# 1 Problématique TOSCA

- Nombre de modules :  $N_{\text{Mod}} = 4$
- Nombre de demandes :  $N_{\text{Dem}} = 5$
- Coûts d'immobilisation :  $I^m = (0.7 \quad 0.4 \quad 0.4 \quad 0.1)$
- Coûts de production :  $P^m = (1.4 \quad 1.8 \quad 1.6 \quad 1.4)$
- Quantités de demandes + Demandes :

$$Q_d = \begin{pmatrix} 7 \\ 6 \\ 3 \\ 1 \\ 24 \end{pmatrix} \quad D_d^m = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

- A nombre fixé de références, quelles sont les **références optimales** minimisant le coût de standardisation ?



# Plan de la présentation

- 1 Problématique TOSCA
- 2 Méthode de résolution actuelle
- 3 Le voisinage pour le recuit simulé
- 4 Perspectives

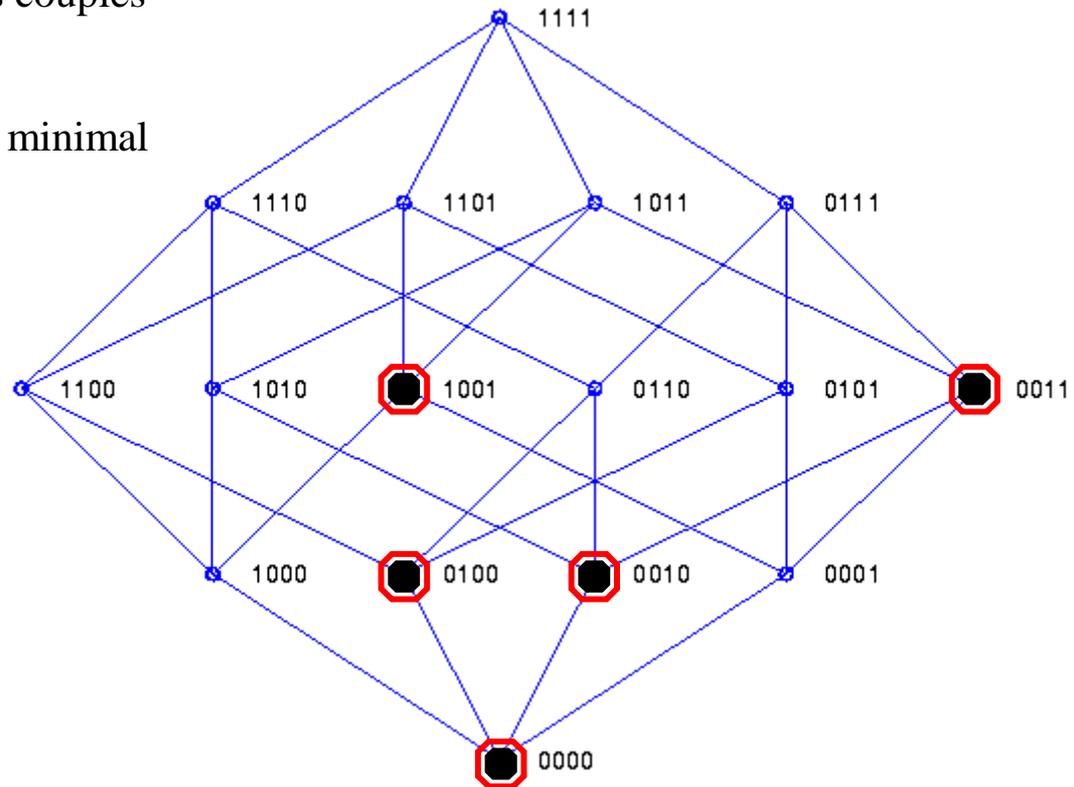


## 2 Méthode de résolution actuelle algorithme glouton

- Recherche incrémentale de la solution :
  - listing des fusions de tous les couples de références
  - choix de la fusion de surcoût minimal

$$R = D = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Algorithme glouton: algo qui construit séquentiellement une solution sans revenir sur sa décision

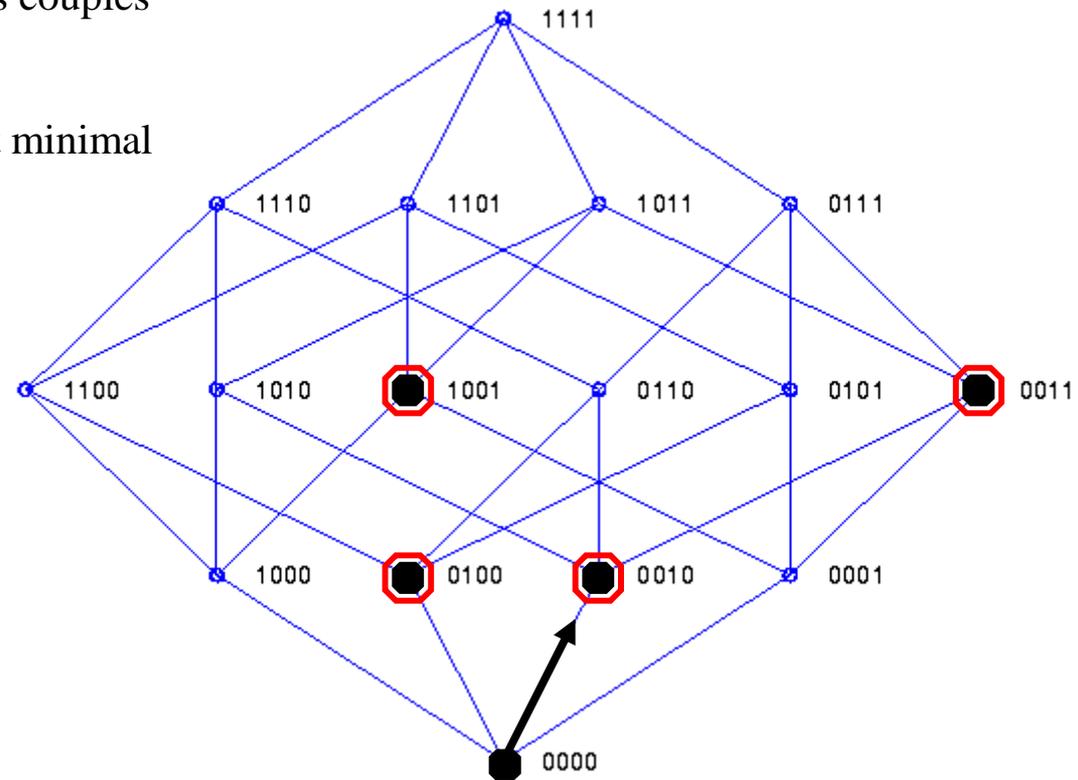




## 2 Méthode de résolution actuelle algorithme glouton

- Recherche incrémentale de la solution :
  - listing des fusions de tous les couples de références
  - choix de la fusion de surcoût minimal

$$R = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ * & * & * & * \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

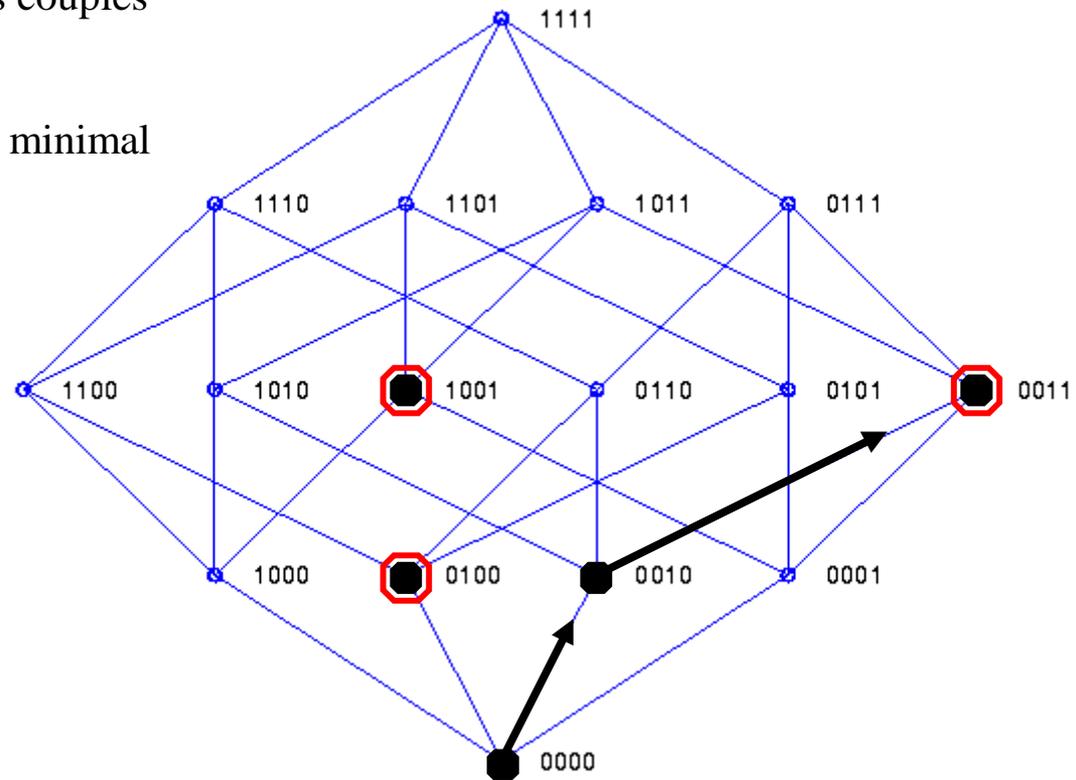




## 2 Méthode de résolution actuelle algorithme glouton

- Recherche incrémentale de la solution :
  - listing des fusions de tous les couples de références
  - choix de la fusion de surcoût minimal

$$R = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ * & * & * & * \\ * & * & * & * \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

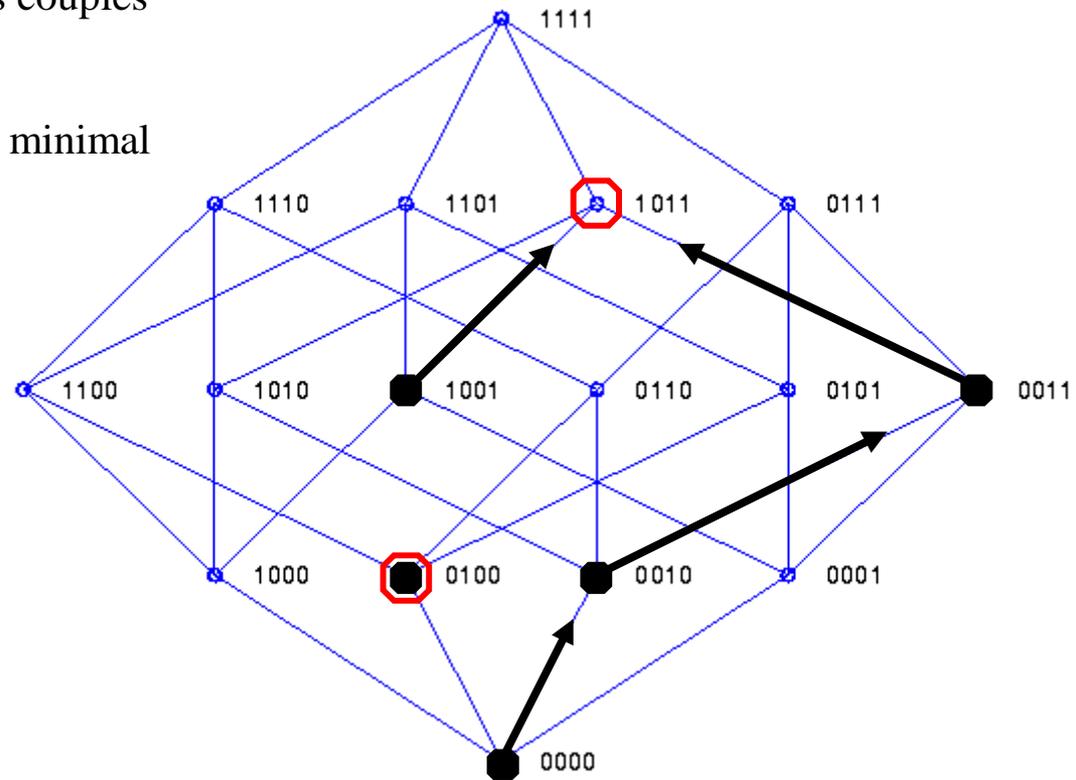




## 2 Méthode de résolution actuelle algorithme glouton

- Recherche incrémentale de la solution :
  - listing des fusions de tous les couples de références
  - choix de la fusion de surcoût minimal

$$R = \begin{pmatrix} 1 & 0 & 1 & 1 \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

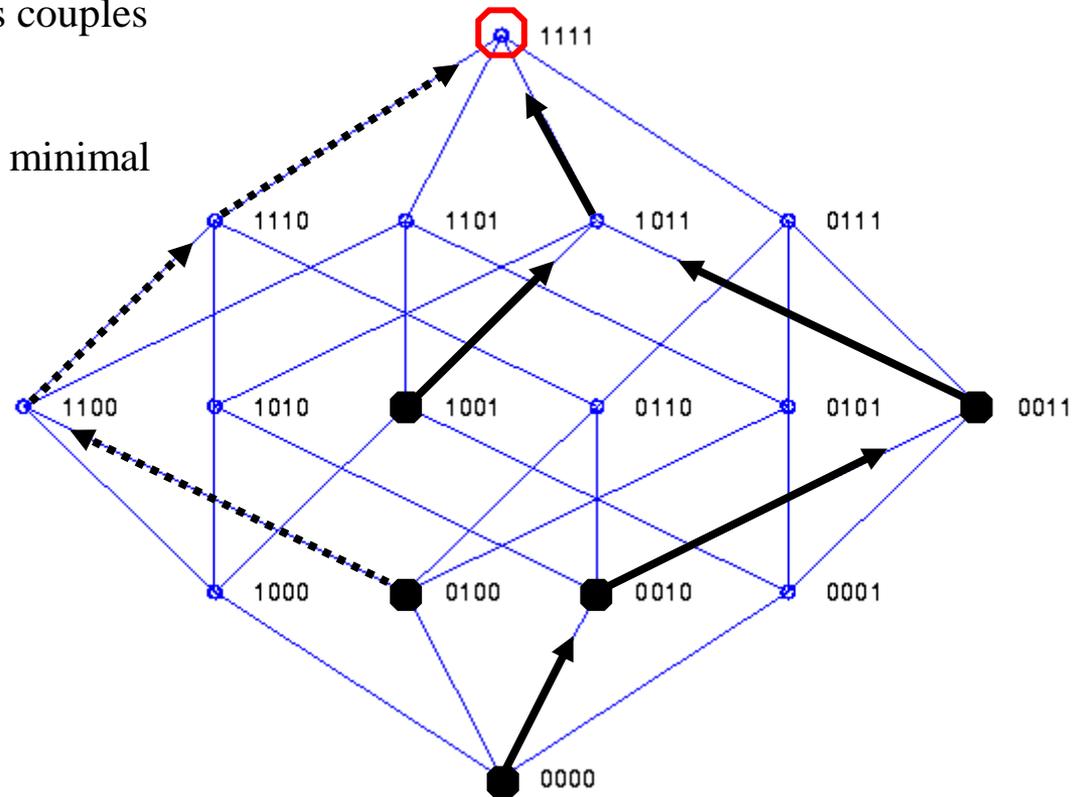




## 2 Méthode de résolution actuelle algorithme glouton

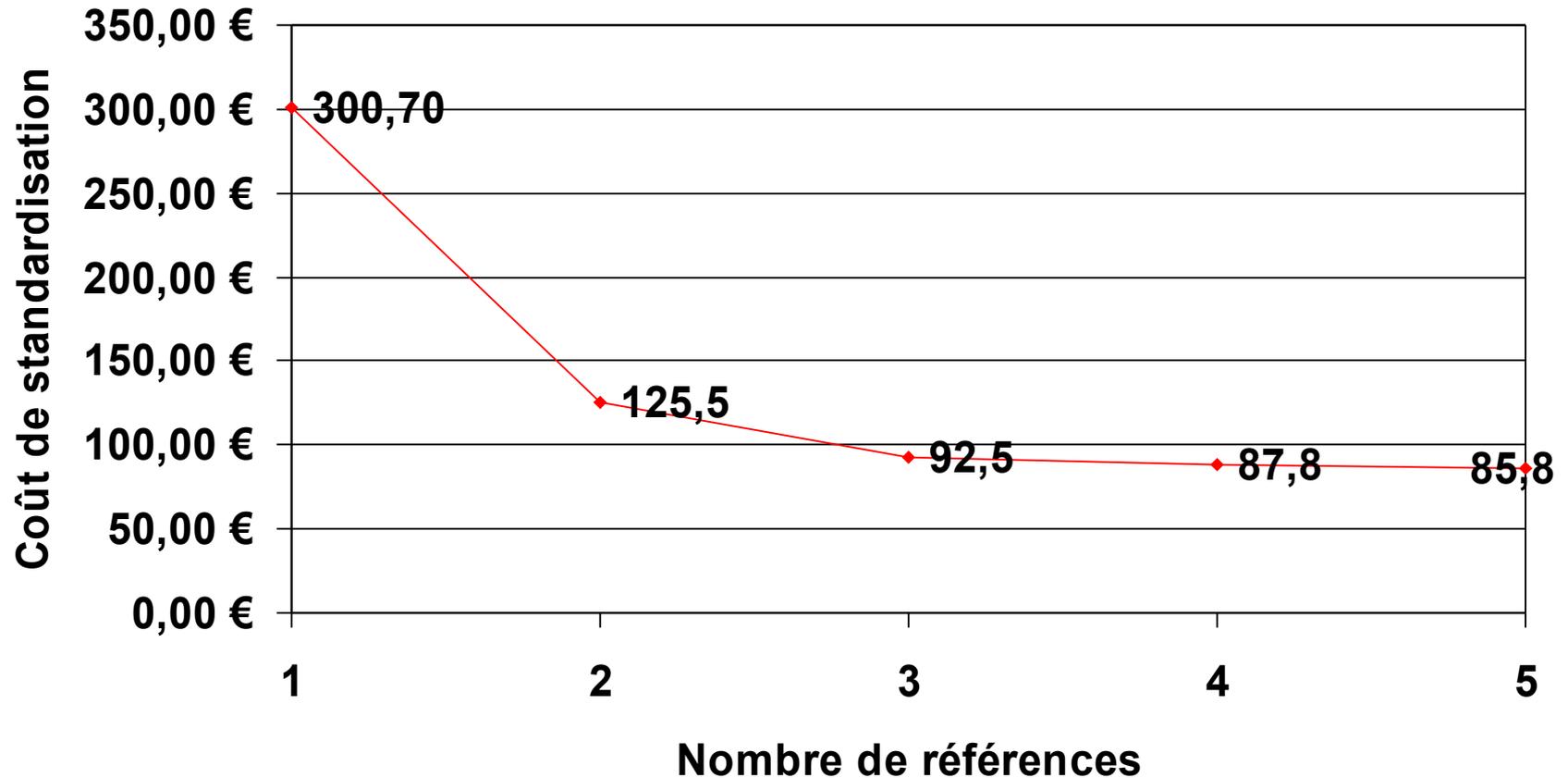
- Recherche incrémentale de la solution :
  - listing des fusions de tous les couples de références
  - choix de la fusion de surcoût minimal

$$R = \begin{pmatrix} 1 & 1 & 1 & 1 \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{pmatrix}$$





## 2 Méthode de résolution actuelle algorithme glouton





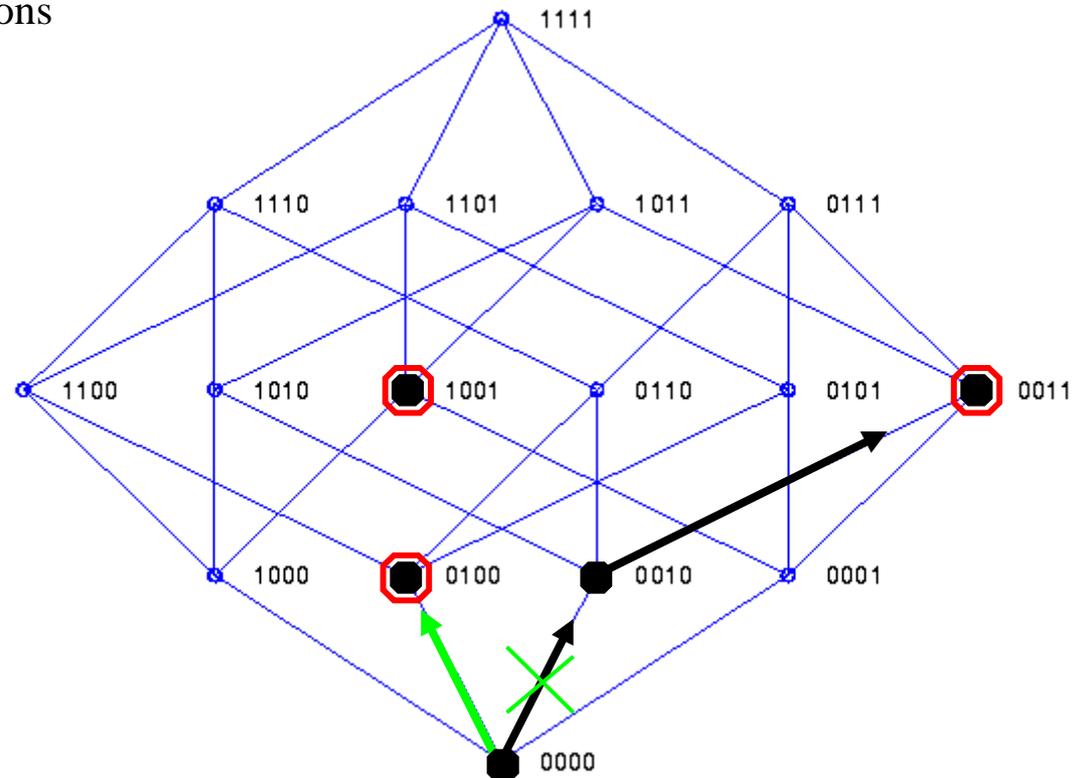
## 2 Méthode de résolution actuelle les limites

- Pas de réaffectation régulière des demandes aux références

=> approximative du surcoût à chaque itération

=> choix non optimal des fusions

=> dégradation de la solution

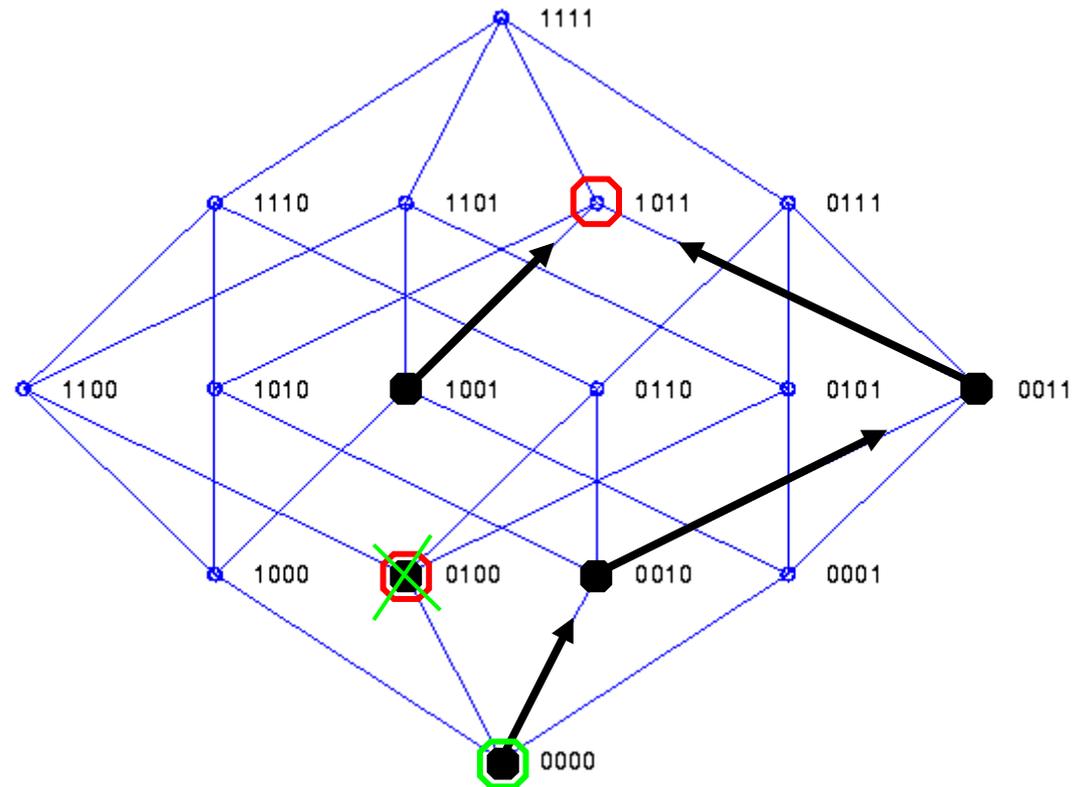




## 2 Méthode de résolution actuelle les limites

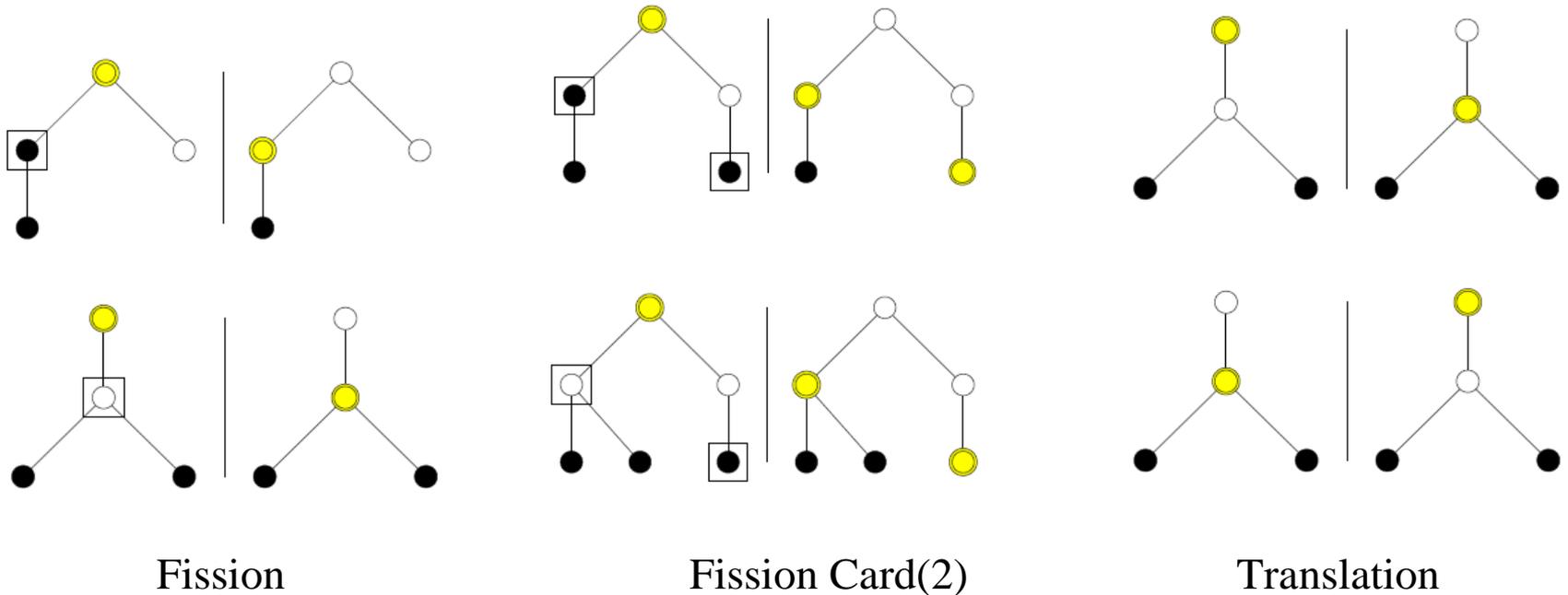
- **Construction itérative** de la solution

=> irréversibilité d'une fusion (une référence supprimée à une itération ne pourra plus jamais être sélectionnée ensuite).





# TOSCA : Le voisinage



Fusion



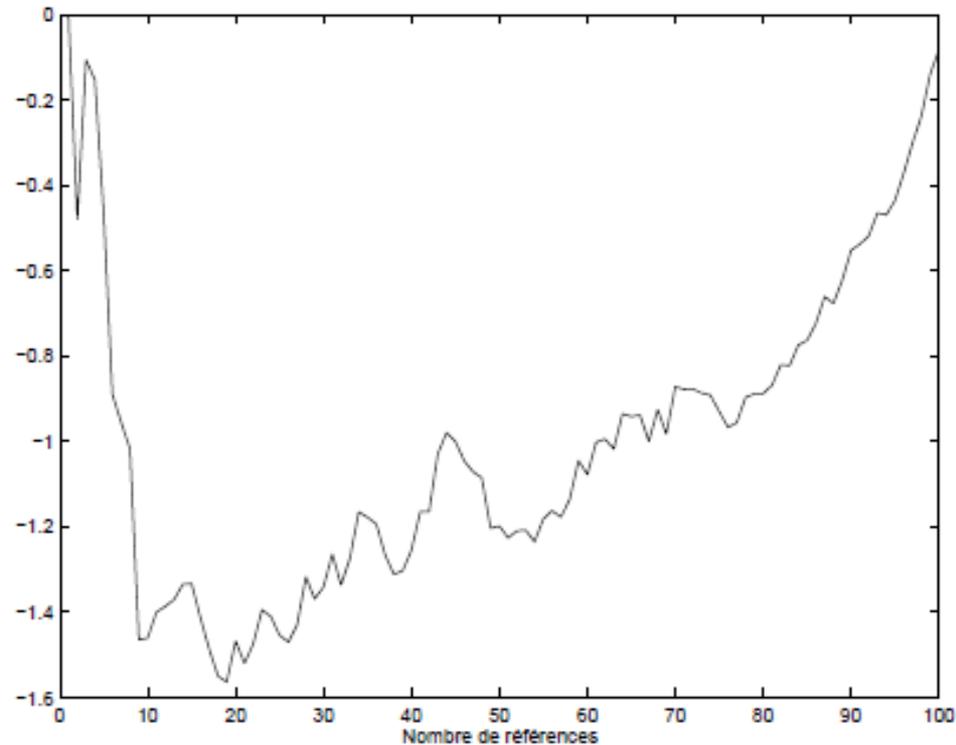
# TOSCA: Le recuit simulé

- Recuit simulé basé sur les précédents mouvements
- La température initiale vaut 0 !
- Le recuit est appliqué à la fin de la passe de l'algorithme glouton



# TOSCA : Résultat

Gain relatif par rapport à la méthode actuelle





# Génération de Cartographies pour calculateur

La génération de cartographies perturbées



# Le système de contrôle : exemple diesel

Calculateur ABS / ESP

Boutonerie régulateur vitesse

Calculateur CLIM

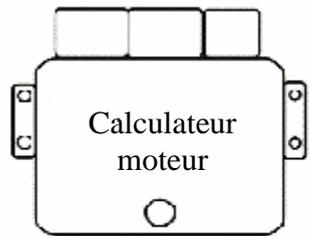
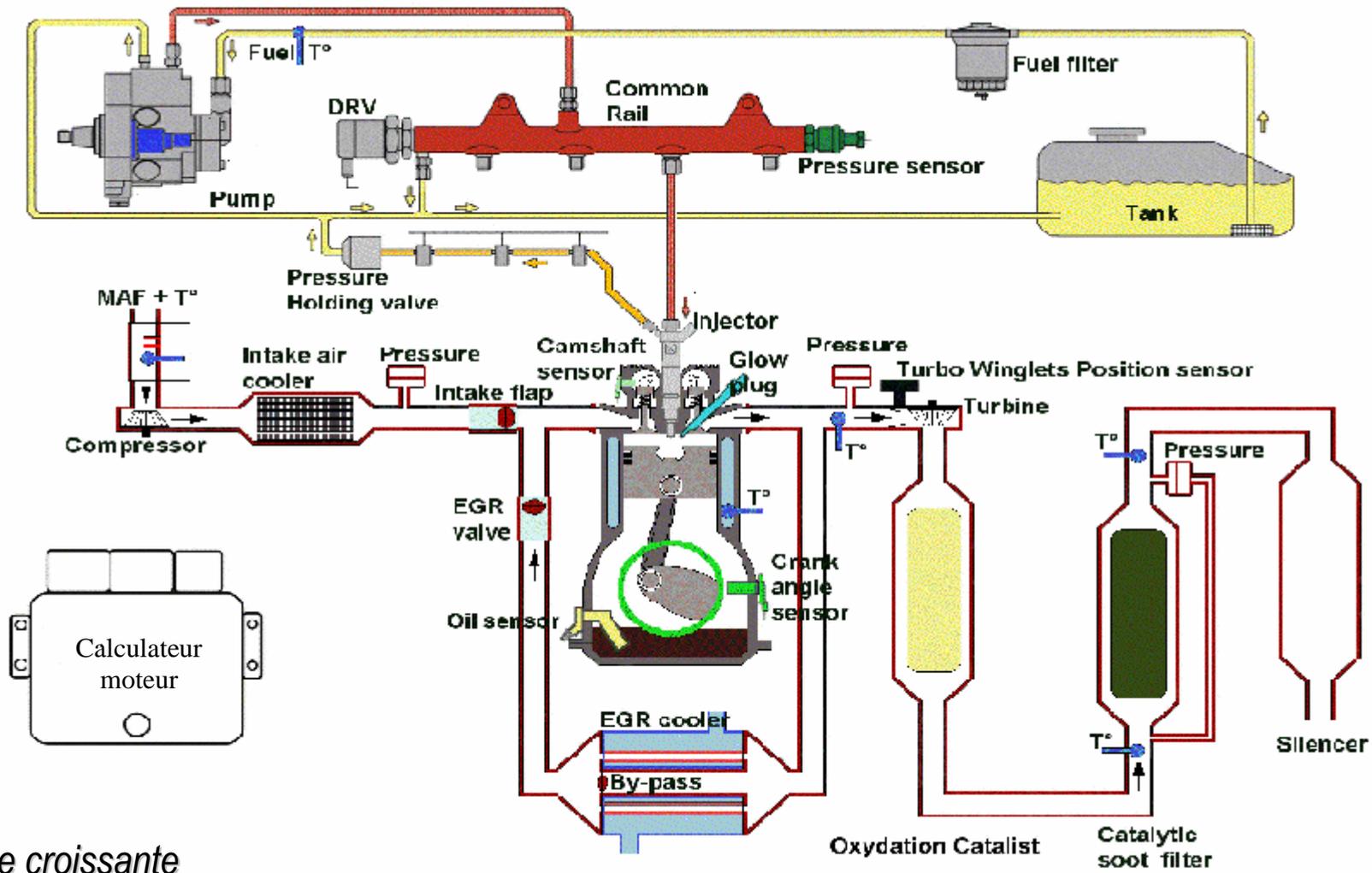
Calculateur BVA

VERLOG

ACC

Tableau de bord ADAC

*Une importance croissante de l'intersystèmes*





# Complexité croissante et impact calibrations

Nb of calibration parameters

+

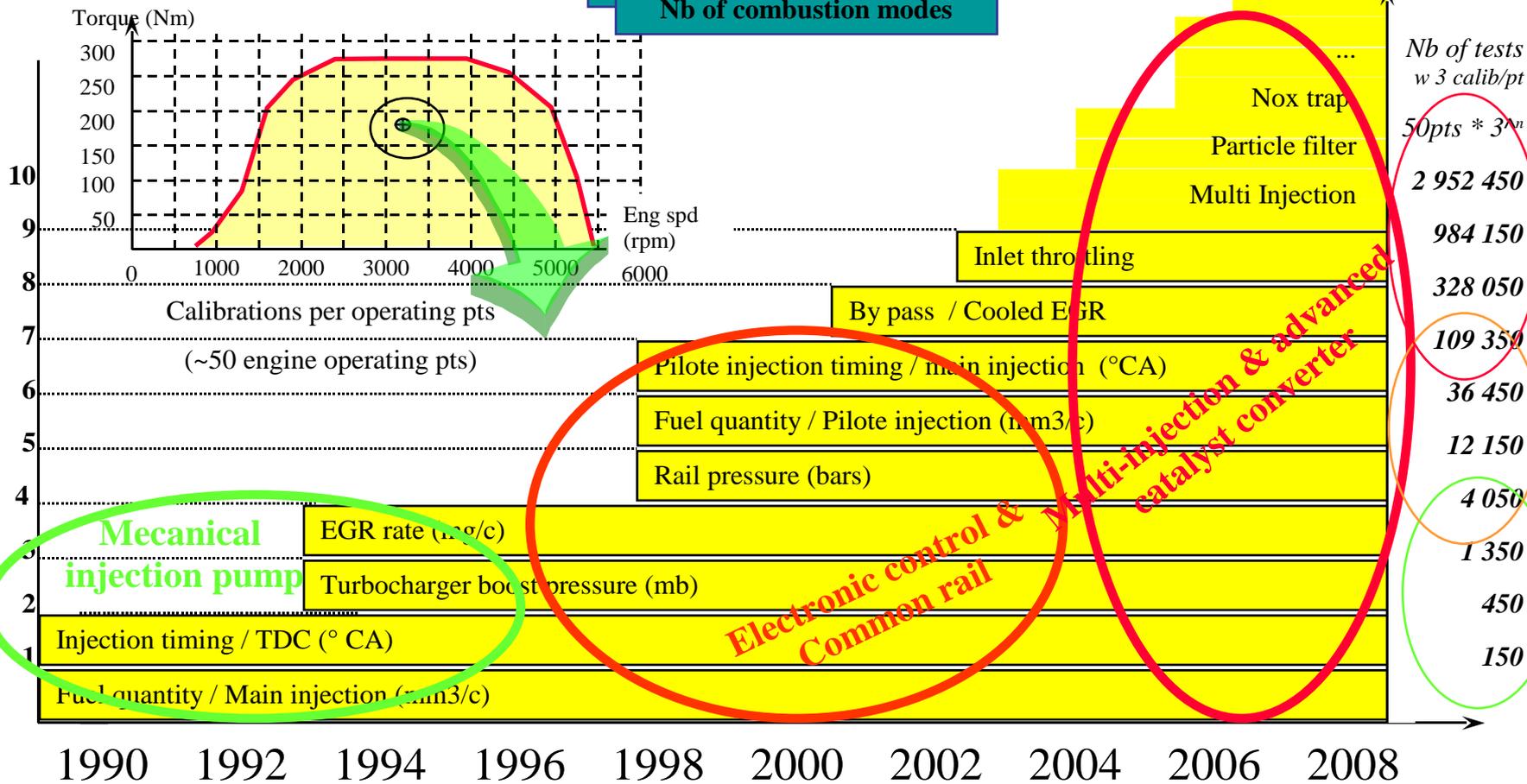
Nb of combustion modes

Nb of combustion modes

Nb of combustion modes

=>

Nb of engine test points



**Mecanical injection pump**

**Electronic control & Common rail**

**Multi-injection & advanced catalyst converter**

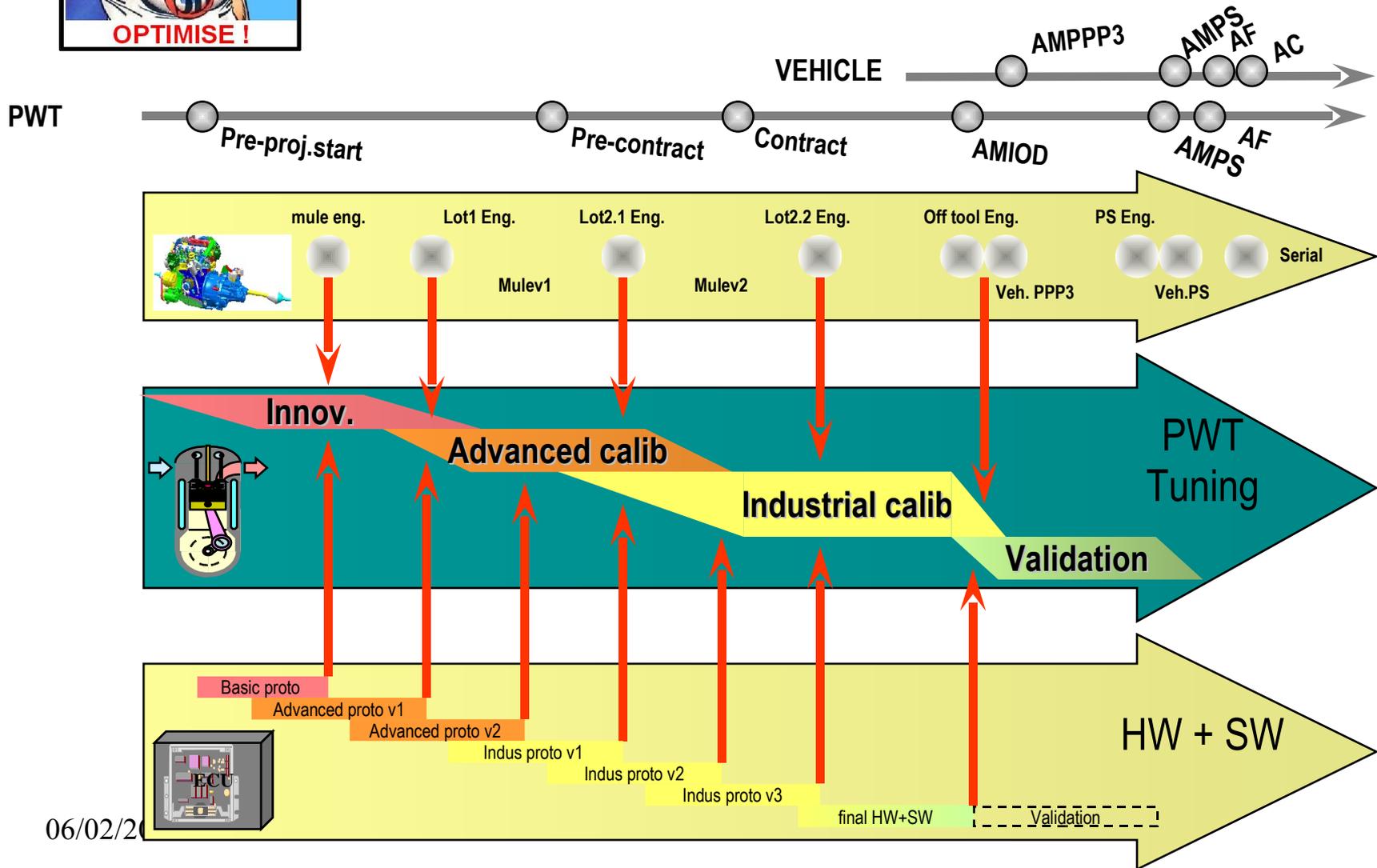
Classical method

Computing (DoE, models)





# Tuning process





# General Process Diesel emission optimisation

Steady state calibrations  
DoE

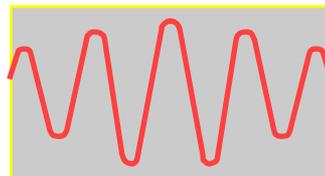
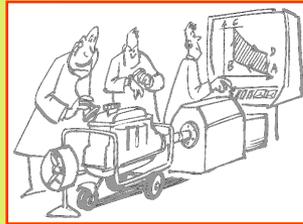
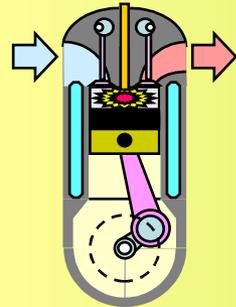
Validation & certification

Dynamic calibrations

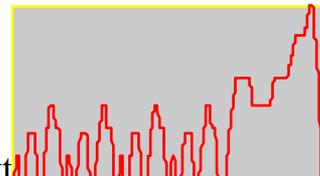
Cold transient (20°C => 90°C)

Warm engine (90°C)

Customer conditions



Collett



time  
65



# Le Banc Moteur



06/02/2006

Y. Collette



# La génération de cartographies perturbées

**But :** Pour un réglage donné, générer un ensemble de  $N$  cartographies couvrant un domaine compris entre une cartographie de borne min et une cartographie de borne max

**Solution :** Utilisation d'un algorithme génétique multiobjectif

**Fonction objectif 1 :** maximiser le minimum d'entropie sur l'ensemble des PF

**Fonction objectif 2 :** maximiser le minimum de variance sur l'ensemble des cartographies

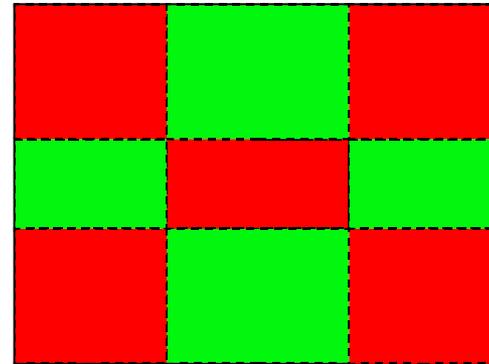
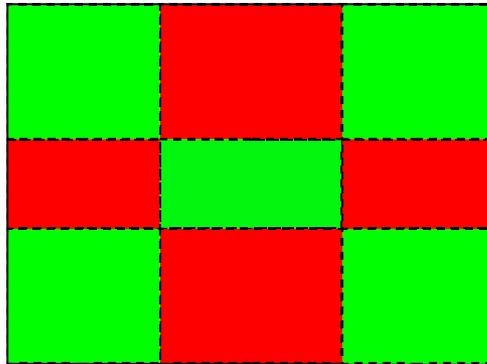
**Contrainte 1 :** sur l'ensemble des cartographies et pour chaque PF, on doit retrouver la borne min et la borne max

**Contrainte 2 :** contrainte de lissage respectée



# Les algorithmes génétiques

## Le croisement de deux cartographies



$$\begin{aligned}\text{Green} &= (1-A).\text{Green} + A.\text{Red} \\ \text{Red} &= A.\text{Green} + (1-A).\text{Red}\end{aligned}$$

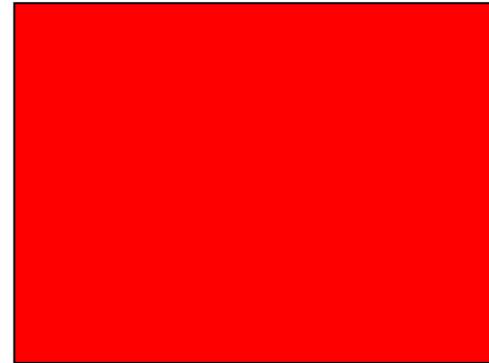
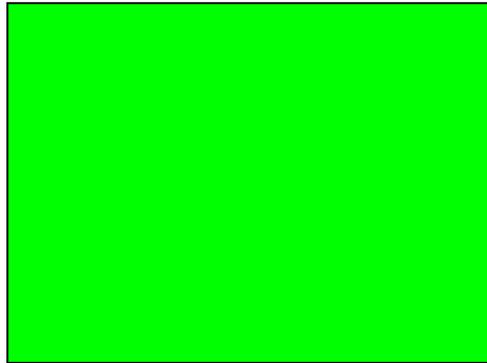
$$-1 < A < 2$$

Corrections:  
Vérification contrainte de lissage  
Vérification contrainte de bornes



# Les algorithmes génétiques

## Le croisement de deux cartographies



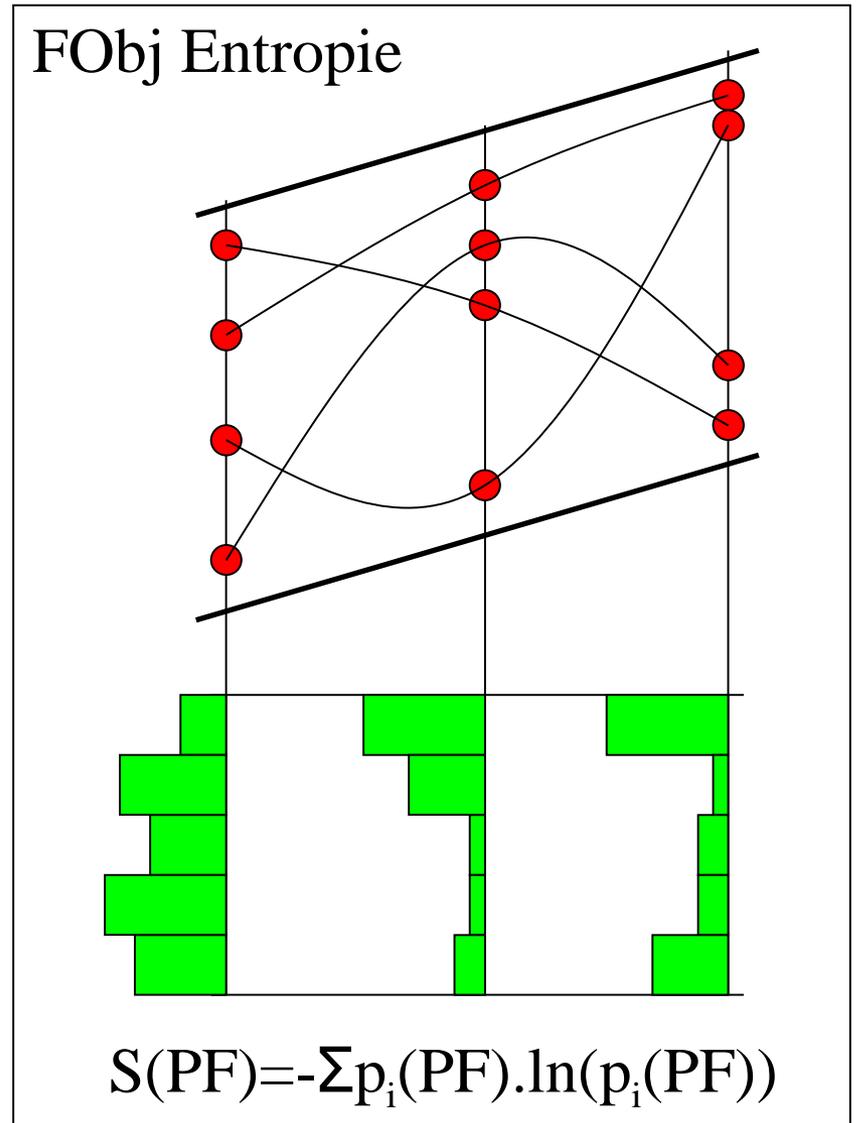
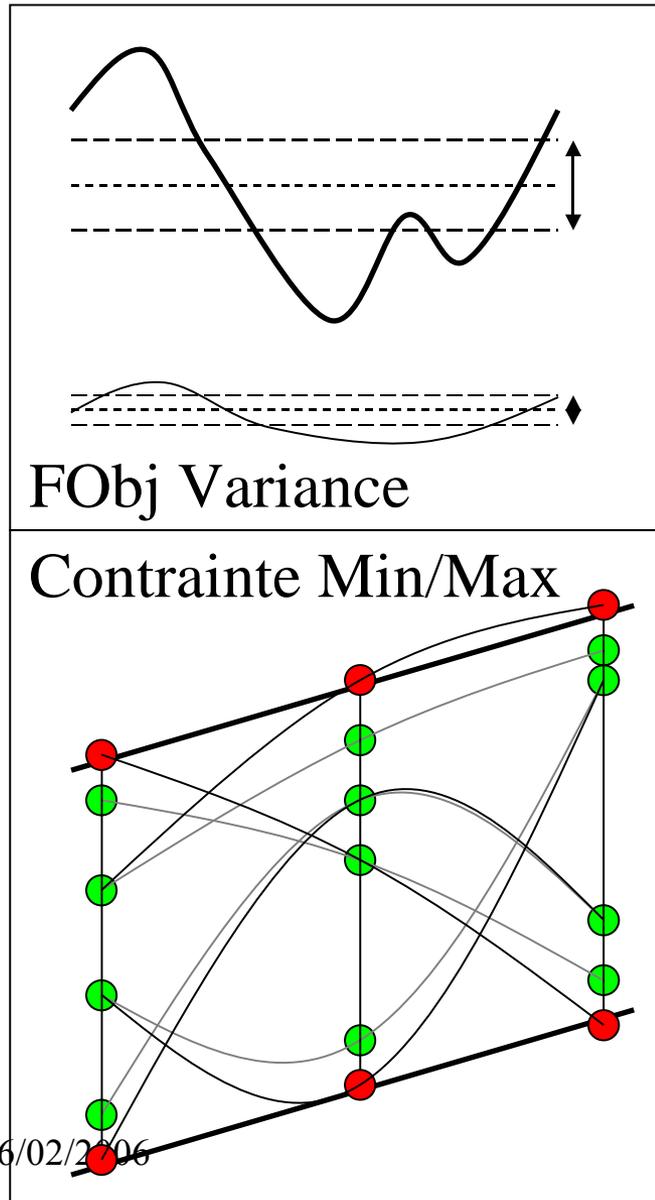
$$\begin{aligned}\text{Green} &= (1-A).\text{Green} + A.\text{Red} \\ \text{Red} &= A.\text{Green} + (1-A).\text{Red}\end{aligned}$$

$$0 < A < 1$$

Correction:  
Vérification contrainte de bornes

Avantage:  
Si la population initiale est lisse  
Les enfants seront lisses

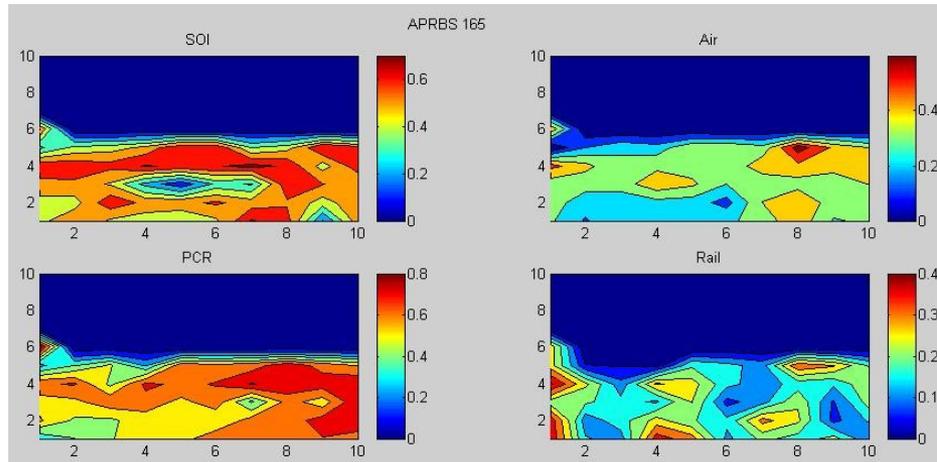
# La génération de cartographies perturbées





# La génération de cartographies perturbées

APRBS



MAPDYN

