Projet : Programmation par contraintes

Set Constraint Modeling - SGP / STS

Jonathan Fontaine, Arthur Gontier, Oryan Rampon, Corentin Pelhâtre, Mathis Ocquident, Thaddeus Leonard, Adrien Cassaigne

UNIVERSITÉ DE NANTES

M2 Informatique ORO

19 décembre 2019

Plan

Plan

Équipe Erasmus :

Groupe de 7 étudiants Erasmus qui n'ont pas eu ce cours l'année dernière

$\Longrightarrow$ faire de son mieux.

Social Golfer Problem :

L'objectif du SGP est de planifier g*p golfeurs dans g groupes de p joueurs sur w semaines tel que :

$\Longrightarrow$ Il n'y a jamais deux golfeurs qui jouent dans le même groupe plus d'une fois.

$\Longrightarrow$ Tous les golfeurs jouent une fois par semaine.

$\Longrightarrow$ Le nombre de groupes et le nombre de joueurs par groupe reste inchangé au cours des w semaines.

Sports Tournament Scheduling :

Le STS est un problème visant a créer un tournois entre n équipes réparties sur n/2 terrains en n-1 semaines.

$\Longrightarrow$ Chaque équipe doit jouer contre les n-1 autres équipes dans le temps imparti.

$\Longrightarrow$ Chaque équipe ne peut jouer qu'au maximum deux fois sur un terrain.

But : fournir un emplois du temps des n/2 terrains sur n-1 semaines avec pour chaque élément de ce dernier, les deux équipes qui se rencontrent.

Plan

Modèle de contraintes ensemblistes

Définition des ensembles

P : l'ensemble des joueurs

S : l'ensemble des semaines

$s \in S$: l'ensemble des groupes d'une semaine

$g \in s \in S, g \subset P$: l'ensemble des joueurs d'un groupe d'une semaine

Modèle de contraintes ensemblistes

Contraintes

Chaque joueur est attribué à un unique groupe par semaine

$$g_1 \cap g_2 = \emptyset \qquad (\forall s \in S)(\forall g_1, g_2 \in s | g_1 \neq g_2)$$

Chaque joueur rencontre les autres au plus une fois

$$|g_1 \cap g_2| \leq 1 \qquad (\forall s_1, s_2 \in S | s_1 \neq s_2)(\forall g_1 \in s_1, \forall g_2 \in s_2)$$

Modèle FD entier :

Variables

$$x_{ijk} = \begin{cases} 1 : \text{Si le joueur k joue dans le groupe j la semaine i.} \\ 0 : \text{Sinon.} \end{cases}$$

$$z_{ijq_1q_2} = \begin{cases} 1 : \text{si } q_1 \text{ et } q_2 \text{ jouent ensemble dans le groupe j la semaine i.} \\ 0 : \text{Sinon.} \end{cases}$$

Modèle FD entier :

Contraintes

$$\sum_{j \in g} x_{ijk} = 1 \qquad \forall i \in w, \forall k \in p \qquad (1)$$

$$\sum_{k \in p} x_{ijk} = p \qquad \forall i \in w, \forall j \in g \qquad (2)$$

$$z_{ijq_1q_2} \geq x_{ijq_1} + x_{ijq_2} - 1 \qquad \forall i \in w, \forall j \in g, \forall q_1, q_2 \in p, q_1 \neq q_2 \qquad (3)$$

$$\sum_{i \in w, j \in g} z_{ijq_1q_2} \leq 1 \qquad \forall q_1, q_2 \in p, q_1 \neq q_2 \qquad (4)$$

Passage de FD entier à SAT : fonction annexe

Algorithme 1 : enumar(f,ii1,jj,k,n,sign)

```
ar = collect(1 :k)
writear(f,ii1,jj,1,k,ar)
pour i in 2 :binomial(n,k) faire
    ii = 0
    tant que ii < k faire
        Si ar[end-ii] + 1≤ n − ii alors
            ar[end-ii] = ar[end-ii] + 1
            pour iii in k-ii+1 :k faire
                ar[iii] = ar[iii-1]+1
            fin
            ii = k
        Sinon
            ii = ii + 1
        fin
    fin
    Si sign alors
        writear(f,ii1,jj,i,k,ar)
    Sinon
        writearneg(f,ii1,jj,i,k,ar)
    fin
fin
```

Exemple d'exécution sur 3 parmi 5 :

1 2 3 - 1 2 4 - 1 2 5 - 1 3 4 - 1 3 5 - 1 4 5 - 2 3 4 - 2 4 5 - 3 4 5

Passage de FD entier à SAT :

la contrainte en "$\geq$" :

$$\sum_{a \in A} x_{ab} \geq c \qquad \forall b \in B \tag{5}$$

$$\Rightarrow \bigwedge_{b \in B} \bigvee_{a \in \binom{q}{q-c+1}} x_{ab} \tag{6}$$

la contrainte en "$\leq$" :

$$\sum_{a \in A} x_{ab} \leq c \qquad \forall b \in B \tag{7}$$

$$\Rightarrow \bigwedge_{b \in B} \Big(\bigvee_{\forall a \in \binom{q}{c+1}} \neg x_{ab} \Big) \tag{8}$$

Modèle SAT (1) :

Contrainte du FD associée :

$$\sum_{j \in g} x_{ijk} = 1 \qquad \forall i \in w, \forall k \in p \tag{9}$$

SAT :

$$\bigwedge_{i,k} \bigvee_{j} x_{ijk} \tag{10}$$

$$\bigwedge_{i,j,j',k;j \neq j'} (\neg x_{ijk} \vee \neg x_{ij'k}) \tag{11}$$

Modèle SAT (2) :

Contrainte du FD associée :

$$\sum_{k \in p} x_{ijk} = p \qquad \forall i \in w, \forall j \in g \tag{12}$$

SAT :

$$\bigwedge_{i,j} \bigvee_{k \in \binom{q}{q-p+1}} x_{ijk} \tag{13}$$

$$\bigwedge_{i,j} \bigvee_{k \in \binom{q}{p+1}} \neg x_{ijk} \tag{14}$$

Modèle SAT (3) :

Contrainte du FD associée :

$$z_{ijq_1q_2} \geq x_{ijq_1} + x_{ijq_2} - 1 \qquad \forall i \in w, \forall j \in g, \forall q_1, q_2 \in p, q_1 \neq q_2 \tag{15}$$

$$\sum_{i \in w, j \in g} z_{ijq_1q_2} \leq 1 \qquad \forall q_1, q_2 \in p, q_1 \neq q_2 \tag{16}$$

SAT :

$$\bigwedge_{(i,j) \neq (i',j'), k \neq k'} (\neg x_{ijk} \vee \neg x_{ijk'} \vee \neg x_{i'j'k} \vee \neg x_{i'j'k'}) \tag{17}$$

Plan

Modèle de contraintes ensemblistes

Définition des ensembles

E : l'ensemble des équipes

$M_{ij} \subset E$: Terrain j semaine i, un ensemble de deux équipes

Modèle de contraintes ensemblistes

Contraintes

Chaque équipe ne joue qu'une fois par semaine :

$$M_{i,j} \cap M_{i,j'} = \emptyset \qquad \forall i, j, j' \ avec \ j \neq j'$$

Chaque équipe joue au plus deux fois sur le terrain j :

$$|M_{i_1 j} \cap M_{i_2 j} \cap M_{i_3 j}| = 0 \qquad \forall i_1 \neq i_2 \neq i_3 \ \ \forall j$$

Cardinal de l'union de toutes les semaines = nombre de match possibles = $(n-1)n/2$

$$|\bigcup_i M_i| = \frac{(n-1)n}{2} \qquad \Leftrightarrow \qquad allDifferent(M_{ij} \forall i, j)$$

Modèle FD entier :

Variables

$$x_{ijk} = \begin{cases} 1 : \text{Si l'équipe k joue sur le terrain j la semaine i.} \\ 0 : \text{Sinon.} \end{cases}$$

$$z_{ijk_1k_2} = \begin{cases} 1 : \text{si } k_1 \text{ et } k_2 \text{ jouent ensemble sur le terrain j la semaine i.} \\ 0 : \text{Sinon.} \end{cases}$$

Modèle FD entier :

Contraintes

Deux équipes sur chaque terrain :

$$\sum_k x_{ijk} = 2 \qquad \forall i, j$$

Chaque équipe joue une fois par semaine :

$$\sum_j x_{ijk} = 1 \qquad \forall i \forall k$$

Chaque équipe joue au plus deux fois sur le même terrain :

$$\sum_i x_{ijk} \leq 2 \qquad \forall j, k$$

Modèle FD entier :

Contraintes

Toutes les équipes se rencontrent au moins une fois

$$z_{ijk_1k_2} \leq \frac{x_{ijk_1} + x_{ijk_2}}{2} \qquad \forall i, j, k_1, k_2 \quad k_1 \neq k_2 \tag{18}$$

$$\sum_{i,j} z_{ijk_1k_2} \geq 1 \qquad \forall k_1, k_2 \quad k_1 \neq k_2 \tag{19}$$

Modèle SAT (1) :

Contrainte du FD associée :

$$\sum_{k} x_{ijk} = 2 \qquad \forall i, j \tag{20}$$

SAT :

$$\bigwedge_{i,j} \bigvee_{k \in \binom{q}{q-1}} x_{ijk} \tag{21}$$

$$\bigwedge_{i,j,k,k',k'';k \neq k' \neq k''} (\neg x_{ijk} \vee \neg x_{ijk'} \vee \neg x_{ijk''}) \tag{22}$$

Modèle SAT (2) :

Contrainte du FD associée :

$$\sum_{j} x_{ijk} = 1 \qquad \forall i \forall k \tag{23}$$

SAT :

$$\bigwedge_{i,k} \bigvee_{j} x_{ijk} \tag{24}$$

$$\bigwedge_{i,j,j',k;j \neq j'} (\neg x_{ijk} \vee \neg x_{ij'k}) \tag{25}$$

Modèle SAT (3) :

Contrainte du FD associée :

$$\sum_{i} x_{ijk} \leq 2 \qquad \forall j, k \tag{26}$$

SAT :

$$\bigwedge_{i,i',i'',j,k;i \neq i' \neq i''} (\neg x_{ijk} \vee \neg x_{i'jk} \vee \neg x_{i''jk}) \tag{27}$$

Modèle SAT (4) :

Contrainte du FD associée :

$$z_{ijk_1k_2} \leq \frac{x_{ijk_1} + x_{ijk_2}}{2} \qquad \forall i, j, k_1, k_2 \quad k_1 \neq k_2 \qquad (28)$$

$$\sum_{i,j} z_{ijk_1k_2} \geq 1 \qquad \forall k_1, k_2 \quad k_1 \neq k_2 \qquad (29)$$

SAT :

$$\bigwedge_{i,j,k,k';k\neq k'} (z_{ijkk'} \vee \neg x_{ijk} \vee \neg x_{ijk'}) \qquad (30)$$

$$\bigwedge_{i,j,k,k';k\neq k'} (\neg z_{ijkk'} \vee x_{ijk}) \qquad \bigwedge_{i,j,k,k';k\neq k'} (\neg z_{ijkk'} \vee x_{ijk'}) \qquad (31)$$

$$\bigwedge_{k,k',k\neq k'} \bigvee_{i,j} z_{ijkk'} \qquad (32)$$

Plan

Symetrie sur les indices

Observation de nombreuse solutions équivalentes dû aux indices
Adaptation : $k \neq k'$ par $k < k'$, $(i,j) \neq (i',j')$
Exemple : $\bigwedge_{k,k',k \neq k'}$ avec $k \in \{1,2,3,4\}$

On a 12 arrangements possibles avec la première modélisation, alors qu'on réduit à 6 en utilisant la seconde tout en conservant toutes les solutions possibles

Première semaine

Algo simple pour fixer la première semaine dans SGP et STS
$\longrightarrow$ casser les symétries améliorer les temps de résolution sur tous les modèles (SAT, FD, ensembliste)

Example

	Sans	Avec
SGP(4,4,4) FD	30,37s	8,13s
SGP(4,4,4) SAT	2,26s	0,07s
SGP(5,5,5) B&B	>5min	1,15s

	Sans	Avec
STS(6) B&B	0.3s	1s
STS(8) B&B	85s	>5min
STS(8) FD	4s	3,8s
STS(10) SAT	0,63s	0,04s

Symetrie sur le modèle ensembliste

Symetries sur le probleme SGP :

On fixe les éléments des groupes de la première semaine

On fixe les éléments du premier groupe de la deuxième semaine

On fixe les premiers éléments des groupes de chaques semaines

Symetrie sur le probleme STS :

On fixe les éléments des terrains de la première semaine

Plan

Filtrage

contrainte : $(v_1 \cap v_2) = \emptyset$

$v_1^{\uparrow} \leftarrow v_1^{\uparrow} \backslash v_2^{\downarrow}$
$v_2^{\uparrow} \leftarrow v_2^{\uparrow} \backslash v_1^{\downarrow}$

contrainte : $|v_1 \cap v_2| \leq 1$

Ne filtrer que si $(v_1 \cap v_2) \neq \emptyset$. Prenons $val \in (v_1 \cap v_2)$:
$v_1^{\uparrow} \leftarrow v_1^{\uparrow} \backslash (v_2^{\downarrow} \backslash val)$
$v_2^{\uparrow} \leftarrow v_2^{\uparrow} \backslash (v_1^{\downarrow} \backslash val)$

contrainte : $(v_1 \cap v_2 \cap v_3) = \emptyset$

$v_1^{\uparrow} \leftarrow v_1^{\uparrow} \backslash (v_2^{\downarrow} \cap v_3^{\downarrow})$
$v_2^{\uparrow} \leftarrow v_2^{\uparrow} \backslash (v_1^{\downarrow} \cap v_3^{\downarrow})$
$v_3^{\uparrow} \leftarrow v_3^{\uparrow} \backslash (v_1^{\downarrow} \cap v_2^{\downarrow})$

structures

Variables

min : l'ensemble minimum de la variable

max : l'ensemble maximum de la variable

card_min : le cardinal minimal de la variable

card_max : le cardinal maximal de la variable

univers : l'univers de la variable

Contrainte

indices_argument : la liste des indices des variables

filtrage ! : l'algorithme de filtrage à appliquer au variables

Algorithme 2 : solver_generique !(variables, contraintes)

```
contraintesVariables ← Tableau qui associe, à la case i, la liste des
  indices des contraintes dans lesquelles est présente la variable à
  l'indice i dans le tableau variables
pileF ← Pile contenant tous les indices des contraintes.
    // contrainte à filtrer
infaisable ← Faux
tant que nonVide(pileF) et non infaisable faire
    indiceCtr = depiler(pileF)
    ctr ← contraintes[indiceCtr]
    filtrer !(ctr, variables)
    pour var ∈ variable(ctr, variables) faire
        Si var à changer alors
            Si valide(var) alors
                pour indCtr ∈ contraintesVariables[var] faire
                    Si indCtr ∉ pileF alors
                        empiler(pileF, indCtr)
                    fin
                fin
            Sinon
                infaisable ← Vrai
            fin
        fin
    fin
fin
return non infaisable
```

Algorithme 3 : branch_and_bound !(variables, contraintes)

```
faisable = solver_generique !(variables, contraintes)
Si faisable alors
    nonClot ← liste de variables non clot
    Si non est_vide(nonClot) alors
        trier nonClot dans l'ordre croissant des variables les plus proches
            d'être réalisées // plus petit différence entre la
            taille de l'ensemble min et le cardinal max
        varBranch ← nonClot[1]
        candidat ← Liste(varBranch.max \ varBranch.min)
        ptr ← candidat.premier
        faisableTemp ← false
        tant que ptr ≠ ∅ and non faisableTemp faire
            varCopie ← copie(variables)
            inserer(varCopie[varBranch].min, ptr.val)
            faisableTemp ← branch_and_bound !(varCopie, contraintes)
            Si non faisableTemp alors
                varBranch.max ← varBranch.max \ ptr.val
            fin
            valeur ← ptr.suiv
        fin
        Si faisableTemp alors
            variables ← varCopie
        Sinon
            faisable ← solver_generique !(variables, contraintes)
        fin
    fin
fin
return faisable
```

Améliorations

- Dans le solver générique, on peut détecter les contraintes close, qui sont toujours vrai, et ne plus les ajouter dans la pile. exemple : $(v_1 \cap v_2) = \emptyset$ avec $(v_1^{\uparrow} \cap v_2^{\uparrow}) = \emptyset$ alors on peut désactiver la contrainte.

- Dans le branch an bound, lors d'un nouvel appel du solver générique, on peut ajouter seulement les contraintes ayant la variable modifié dans le branchement.

Plan

Résultats SGP 4-4-4

Ensembliste 4-4-4		
Semaine 1	Groupe 1	[1,2,3,4]
	Groupe 2	[5,6,7,8]
	Groupe 3	[9,10,11,12]
	Groupe 4	[13,14,15,16]
Semaine 2	Groupe 1	[1,5,9,13]
	Groupe 2	[2,7,10,16]
	Groupe 3	[3,8,11,14]
	Groupe 4	[4,6,12,15]
Semaine 3	Groupe 1	[1,7,11,15]
	Groupe 2	[2,5,12,14]
	Groupe 3	[3,6,9,16]
	Groupe 4	[4,8,10,13]
Semaine 4	Groupe 1	[1,6,10,14]
	Groupe 2	[2,8,9,15]
	Groupe 3	[3,7,12,13]
	Groupe 4	[4,5,11,16]

Temps	**0,091 s**

Ensembliste	0,091 s
FD-Gurobi	0.313877 s
SAT-z3	0.04 s
SAT-Minisat	1.6285 s
Branch And Bound	0.077761 s

Résultats SGP 5-5-5

Ensembliste					
Semaine 1	Groupe 1	[1,2,3,4,5]	**Semaine 4**	Groupe 1	[1,8,13,18,23]
	Groupe 2	[6,7,8,9,10]		Groupe 2	[2,6,15,17,24]
	Groupe 3	[11,12,13,14,15]		Groupe 3	[3,10,11,19,22]
	Groupe 4	[16,17,18,19,20]		Groupe 4	[4,9,12,16,25]
	Groupe 5	[21,22,23,24,25]		Groupe 5	[5,7,14,20,21]
Semaine 2	Groupe 1	[1,6,11,16,21]	**Semaine 5**	Groupe 1	[1,7,14,17,22]
	Groupe 2	[2,9,13,20,22]		Groupe 2	[2,10,14,16,23]
	Groupe 3	[3,8,14,17,25]		Groupe 3	[3,9,15,18,21]
	Groupe 4	[4,7,15,19,23]		Groupe 4	[4,8,11,20,24]
	Groupe 5	[5,10,12,18,24]		Groupe 5	[5,6,13,19,25]
Semaine 3	Groupe 1	[1,9,14,19,24]			
	Groupe 2	[2,7,11,18,25]			
	Groupe 3	[3,6,12,20,23]		**Temps**	**0.134 s**
	Groupe 4	[4,10,13,17,21]			
	Groupe 5	[5,8,15,16,22]			

Ensembliste	0,134 s
FD-Gurobi	6.227383 s
SAT-z3	7.00 s
SAT-Minisat	Time Out
Branch And Bound	1.503023 s

Résultats STS 6

FD		
Semaine 1	Terrain 1	[1, 2]
	Terrain 2	[3, 4]
	Terrain 3	[5, 6]
Semaine 2	Terrain 1	[1, 5]
	Terrain 2	[3, 6]
	Terrain 3	[2, 4]
Semaine 3	Terrain 1	[4, 6]
	Terrain 2	[2, 5]
	Terrain 3	[1, 3]
Semaine 4	Terrain 1	[3, 5]
	Terrain 2	[1, 4]
	Terrain 3	[2, 6]
Semaine 5	Terrain 1	[2, 3]
	Terrain 2	[1, 6]
	Terrain 3	[4, 5]
Temps	0.032309 s	

Ensembliste	0.082309 s
FD-Gurobi	0.032309 s
SAT-z3	0.00 s
SAT-Minisat	0.007741 s
Branch And Bound	0.364210

Résultats STS 8

FD					
Semaine 1	Terrain 1	[1, 2]	**Semaine 5**	Terrain 1	[5, 7]
	Terrain 2	[3, 4]		Terrain 2	[1, 3]
	Terrain 3	[5, 6]		Terrain 3	[4, 8]
	Terrain 4	[7, 8]		Terrain 4	[2, 6]
Semaine 2	Terrain 1	[6, 7]	**Semaine 6**	Terrain 1	[1, 8]
	Terrain 2	[5, 8]		Terrain 2	[1, 8]
	Terrain 3	[2, 3]		Terrain 3	[3, 7]
	Terrain 4	[1, 4]		Terrain 4	[4, 6]
Semaine 3	Terrain 1	[3, 6]	**Semaine 7**	Terrain 1	[2, 4]
	Terrain 2	[4, 7]		Terrain 2	[6, 8]
	Terrain 3	[2, 8]		Terrain 3	[1, 7]
	Terrain 4	[1, 5]		Terrain 4	[3, 5]
Semaine 4	Terrain 1	[3, 8]			
	Terrain 2	[1, 6]			
	Terrain 3	[4, 5]			
	Terrain 4	[2, 7]			
		Temps	3.760724 s		

Ensembliste	0.201 s
FD-Gurobi	3.760724 s
SAT-z3	0.01 s
SAT-Minisat	0.032834 s
Branch And Bound	87.801295 s

Résultats STS 10

Ensembliste	1.05 h
FD-Gurobi	Time Out
SAT-z3	0.02 s
SAT-Minisat	0.019324 s
Branch And Bound	Time Out

Résultats STS 12

Ensembliste	Time Out
FD-Gurobi	Time Out
SAT-z3	59.70 s
SAT-Minisat	88.2098 s
Branch And Bound	Time Out

Analyse des Résultats

Analyse

- SGP
  - Problème plus combinatoire que le STS
  - On observe que l'on peut casser plus de symétrie que sur le STS
- STS
  - Problème moins combinatoire SGP
  - Beaucoup de clauses à deux litéraux

Améliorations

- SGP
  - On pourrait modifier le modèle en générant les groupes possibles pour les utiliser comme valeurs de variables ensembliste (il ne faudrait générer que $\frac{n(n-1)}{k(k-1)}$ avec n le nombre de joueurs et k le nombre de joueurs dans un groupe).

Plan

Conclusion :

- Modèle ensembliste est le meilleur sur le SGP.
- Modèle SAT est le meilleur sur le STS.
- Z3 est meilleur que Minisat.
- Notre solveur artisanal talonne minizinc sur le SGP
- Notre solveur artisanal est faible sur le STS a cause de son filtrage