174 Problème du Sudoku (Au tableau)

j

i

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
8	5	9	7	6	1	4	2	3
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

1. Choix des variables

$$\{ x_{i,j,v} \mid i,j,v \in \{1,\dots,9\} \}$$

$x_{i,j,v}$ sera vraie si la case de coordonnées (i,j) contient la valeur v

2. Pas deux fois la même valeur sur une même ligne :

On fixe i (la ligne) et on veut exprimer que "il n'existe pas deux cases avec la même valeur v sur la ligne i"

Fixons une valeur v.

"il n'existe pas j,k tels que $j \neq k$ et $\begin{cases} (i,j) \text{ contient } v \\ \quad\text{et} \\ (i,k) \text{ contient } v \end{cases}$"

"il n'existe pas j et k tels que $j \neq k$ et $\begin{cases} (i,j) & \text{contient } v \text{ "} \\ & \text{et} \\ (i,k) & \text{contient } v \end{cases}$

$$\neg \left(\bigvee_{\substack{j,k \in \{1,\dots,9\} \\ j \neq k}} x_{i,j,v} \wedge x_{i,k,v} \right)$$

$$\equiv \bigwedge_{\substack{j,k \in \{1,\dots,9\} \\ j \neq k}} \left(\neg x_{i,j,v} \vee \neg x_{i,k,v} \right)$$

Il faut le faire pour tout i, v :

$$\boxed{\bigwedge_{i,v \in \{1,\dots,9\}} \bigwedge_{\substack{j,k \in \{1,\dots,9\} \\ j \neq k}} \left(\neg x_{i,j,v} \vee \neg x_{i,k,v} \right)}$$

" Pour toute paire de cases différentes (i,j) et (i,k) et pour toute valeur v, une des deux cases ne contient pas v "

$$\bigwedge_{\substack{i,j,k\in\{1,\dots,9\}\\ v\in\{1,\dots,9\}\\ j\neq k \text{ et } k>j}} \left(\neg X_{i,j,v} \vee \neg X_{i,k,v}\right)$$

pas nécessaire mais évite de considérer deux fois chaque clause

3. Pas deux fois la même valeur sur une même colonne :

$$\bigwedge_{\substack{i,j,k\in\{1,\dots,9\}\\ v\in\{1,\dots,9\}\\ i\neq k \text{ et } k>i}} \left(\neg X_{i,j,v} \vee \neg X_{k,j,v}\right)$$

4. Pas deux fois la même valeur dans le même carré :

$$\bigwedge_{\substack{(i,j) \in \{1,..,9\}^2 \\ (k,l) \in \{1,..,9\}^2 \\ (i,j) \neq (k,l) \\ (i,j) \text{ et } (k,l) \\ \text{sont dans le même} \\ \text{carré } 3\times 3}} \quad \bigwedge_{v \in \{1,..,9\}} \left(\neg x_{i,j,v} \vee \neg x_{k,l,v} \right)$$

(i,j) et (k,l) sont dans le même carré 3x3 $\longrightarrow$ comment tester ?

OPTIMISATION : Prendre $k \geq i$ et $l > j$ ou $k > i$ et $l \geq j$ pour éliminer de la redondance.

(i,j) et (k,l) sont dans le même carré <u>ssi</u>

$(i-1)/3 = (k-1)/3$ et $(j-1)/3 = (l-1)/3$

(division entière)

5. Au moins une valeur par case :

$$\bigwedge_{i,j \in \{1,..,9\}} \left(\bigvee_{v \in \{1,..,9\}} x_{i,j,v} \right)$$

Remarque : la contrainte "au plus une valeur par case" n'est pas nécessaire car elle est impliquée par toutes les contraintes précédentes.

6. Contraintes de la grille de départ :

Si la grille est donnée par une fonction

$$G : \{1,..,9\}^2 \to \{1,..,9\} \cup \{\perp\}$$

↑ cases qui ne contiennent pas de valeur initiale

$$\longrightarrow \bigwedge_{\substack{i,j \in \{1,..,9\} \\ G(i,j) \neq \perp}} x_{i,j,G(i,j)}$$

Questions sur le Sudoku

Discussion

1. comment tester l'unicité d'une solution ?
2. comment générer une grille avec au moins une solution ?
3. comment générer k grilles différentes, chacune avec au moins une solution ?
4. ~~on veut générer des grilles ayant au moins deux solutions. Pour cela on se donne un ensemble de cases qui doivent contenir des valeurs pour la grille de départ, alors que les autres seront à remplir par le joueur. Comment utiliser un SAT solveur pour résoudre ce problème ?~~
5. comment générer des grilles avec une solution unique ?

1. Si $S : \{1, .., 9\}^2 \to \{1, .., 9\}$ est une solution à la grille et si Φ est la formule construite précédement (toutes les contraintes prises en conjonction), alors on a :

$$\Phi \to \bigwedge_{i,j} X_{i,j,S(i,j)} \quad \text{valide si et seulement}$$

si la solution S est unique.

De manière équivalente $\neg\left(\Phi \to \bigwedge_{i,j} X_{i,j,S(i,j)}\right)$ n'est pas sat. ssi S est unique

$\Longrightarrow$ tester avec Minisat la satisfaisabilité de $\Phi \wedge \left(\bigvee_{i,j} \neg X_{i,j,S(i,j)}\right)$

2. Comment générer une grille avec au moins une solution ?

- Appeler le solveur sur une grille vide
- Retirer des éléments de la grille pleine trouvée en faisant un test d'unicité de la solution à chaque fois, autant de fois que cela est possible

- on obtient à la fin une grille partiellement remplie qui a une unique solution

3. k grilles différentes

- Répéter k fois l'algorithme précédent avec les contraintes suivantes :

- Si $S_i : \{1, \dots, 9\}^2 \longrightarrow \{1, \dots, 9\}$ est la grille pleine trouvée à l'étape i, à l'étape $i+1$, appeler le solveur sur la formule

$$\Phi_{i+1} \equiv \Phi_i \wedge \bigvee_{(a,b) \in \{1, \dots, 9\}^2} \neg X_{a,b,S(a,b)}$$

5. Comment générer des grilles à solution unique ?

Une solution a été donnée à la question 2, mais elle fait appel plusieurs fois au solveur SAT. Pourrait-on faire mieux ? Pourrait-on par exemple étant donné un entier k trouver un moyen de générer en un seul appel à SAT une grille avec k valeurs pré-remplies et avec une solution unique ? On aimerait de plus que toute grille avec k valeurs préremplies et solution unique soit générable de cette manière.

Cela semble difficile avec une formule de taille polynomiale en k, car déjà le problème, étant donnée une grille préremplie, de savoir si cette grille a une solution unique ne semble pas pouvoir être résoluble par un seul appel du solveur sur une formule de taille polynomiale : on peut tester l'existence d'une solution, mais pour tester l'unicité, il faudrait pouvoir quantifier sur toutes les interprétations possibles des variables.

Si $\Phi_G((X_{i,j,v})_{i,j,v})$ est la formule construite sur l'ensemble de variables $X_{i,j,v}$ et permettant de tester l'existence d'une solution, alors la formule suivante est satisfaisable si et seulement si il existe une unique solution :

$$\underbrace{\exists (X_{i,j,v})_{i,j,v} \cdot \Phi_G\big((X_{i,j,v})_{i,j,v}\big)}_{\text{au moins une solution}} \wedge \underbrace{\forall (Y_{i,j,v})_{i,j,v} \cdot \Phi_G\big((Y_{i,j,v})_{i,j,v}\big) \longrightarrow \bigwedge_{i,j,v} X_{i,j,v} \leftrightarrow Y_{i,j,v}}_{\text{au plus une solution}}$$