Sudoku

Sujet proposé par Jean-Pierre Tillich

Difficulté : facile(*) à moyenne (**)

Distribué le 29 janvier 2014

1 Quelques mots sur le sujet

Dans ce projet il est d'abord demandé d'implémenter un algorithme de backtracking simple pour résoudre une grille de sudoku. Ensuite il s'agira de comprendre comment ce problème peut se voir comme une instance particulière d'un problème plus général en combinatoire, le *problème de la couverture exacte* et d'implémenter une méthode de résolution de ce dernier problème due à Donald Knuth faisant appel aux listes doublement chaînées et qui se révèle très efficace dans le cas du sudoku. Dans une deuxième partie il est demandé d'utiliser cet algorithme de résolution pour créer des grilles partiellement remplis et qui se complètent de manière unique et enfin de trouver au moyen du programme que vous aurez écrit des sudoku qui se complètent de manière unique de taille minimale.

2 Le sudoku

On considérera ici généralisation carrée du sudoku standard, c'est à dire une grille carrée de taille $n^2 \times n^2$, où n est l'ordre du sudoko ($n = 3$ dans le cas standard), qui se décompose en $n \times n$ blocs carrés de taille $n \times n$, qu'il faut remplir avec les nombres de 1 à n^2 en respectant les règles suivantes :

- Chaque ligne et colonne contient exactement une fois tous les nombres de 1 à n^2,
- chaque bloc carré contient lui aussi exactement une fois tous les nombres de 1 à n^2.

Voici un exemple d'une grille partiellement remplie avec sa solution

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

Le jeu consiste à fournir une grille d'un sudoku valide partiellement remplie et le but est de compléter cette grille en un sudoku valide. Généralement, ces grilles partielles sont constituées de telle manière que la solution finale soit unique. On appelle grille de sudoku *minimale*, une telle grille se complétant en un unique sudoku qui est telle qu'on ne puisse lui enlever un seul nombre sans perdre l'unicité de la solution.

3 Résoudre les sudoku

3.1 Un premier algorithme de backtracking simple

Le problème du sudoku de taille $n^2 \times n^2$ est connu comme étant NP-complet. Cela ne veut pas dire que pour des petites tailles (comme ici le cas qui nous intéresse le plus et qui correspond à $n = 3$) ne peut pas être résolu de manière efficace. On commencera par implémenter un premier algorithme consistant simplement à explorer l'arbre des possibilités consistant à remplir de plus en plus de cases de la grille tout en respectant les contraintes. Quand on se trouve bloqué, c'est à dire qu'aucune nouvelle case ne peut être complétée sans violer les contraintes, on revient au dernier embranchement non encore emprunté dans l'arbre des possibilités et on continue l'exploration à partir de là.

3.2 Le problème de la couverture exacte

On considérera dans un deuxième temps le problème de la résolution d'un sudoku comme un problème de couverture exacte. Ce problème peut se formaliser comme suit. On se donne un ensemble fini E et une collection $\mathcal{S}$ de sous-ensembles de E. Une couverture exacte est un sous-ensemble $\mathcal{S}^*$ de $\mathcal{S}$ tel que tout élément de E appartient à un sous-ensemble unique de $\mathcal{S}^*$. En d'autres termes, $\mathcal{S}^*$ est une partition de E. Le problème de la couverture exacte est de déterminer pour un couple $(E, \mathcal{S})$ donné si une telle couverture exacte existe, et le cas échéant d'en exhiber une (voire toutes les couvertures exactes).

A titre d'exemple, considérons l'exemple suivant.

$$\begin{aligned} E &= \{1, 2, 3, 4\} \\ \mathcal{S} &= \{\{1, 2, 3\}, \{1, 3\}, \{2, 3\}, \{2, 4\}\} \end{aligned}$$

Il y a une seule couverture exacte dans ce cas. Elle est donnée par

$$\mathcal{S}^* = \{\{1, 3\}, \{2, 4\}\}$$

Le problème de la couverture peut être résolu de manière relativement efficace par l'algorithme X de Knuth [2, 1] qui s'implante avec des techniques de liste doublement chainées [2].

Il est demandé de formaliser dans un premier temps le problème de la résolution d'un sudoku comme un problème de couverture exacte, puis d'implanter l'algorithme X de Knuth et de l'utiliser pour résoudre un sudoku.

4 Création de grilles de sudoku

Dans un troisième temps, il est demandé d'écrire un programme créant une grille de sudoku valide puis d'utiliser ce programme ainsi que l'algorithme de résolution précédent pour créer des grilles minimales. On essaiera ici de créer des grilles qui contiennent un nombre minimum de cases pré-remplies. Il a été prouvé récemment que 17 était le plus petit nombre de cases pré-remplies d'une grille minimale, arriverez-vous à trouver une grille de sudoku minimale ayant 17 cases remplies seulement ?

Références

[1] `http://en.wikipedia.org/wiki/Knuth's_Algorithm_X`.

[2] D. Knuth, Donald E. (2000), "Dancing links", dans J. Davies, B. Roscoe, J. Woodcock, Millennial Perspectives in Computer Science, actes de l'Oxford-Microsoft Symposium 1999 en l'honneur de Sir Tony Hoare, Palgrave, pp. 187-214, voir aussi `http://arxiv.org/abs/cs/0011047`.