53921 PARREAU--BOULMÉ Mélissandre

Résolution efficace, générique et validée expérimentalement de sudokus

Je trouve amusant mais surprenamment difficile de construire un programme fonctionnel et efficace résolvant des sudokus de taille quelconque. Pour le mettre au point, je m'appuie sur les mathématiques du sudoku (nombre de solutions).

Tout sudoku est convertible en problème de coloriage de graphe. Ces problèmes sont bien étudiés, à la fois mathématiquement et algorithmiquement : backtracking et propagation de contraintes, une forme de système de transition. Les symétries du sudoku, c'est-à-dire les transformations bijectives préservant les règles, sont aussi un ingrédient mathématique essentiel.

Positionnement thématique (ÉTAPE 1) :

- *INFORMATIQUE (Informatique pratique)*
- *INFORMATIQUE (Informatique Théorique)*
- *MATHEMATIQUES (Mathématiques Appliquées)*

Mots-clés (ÉTAPE 1) :

Mots-clés (en français)	**Mots-clés (en anglais)**
backtracking	*backtracking*
propagation	*propagation*
coloriage de graphes	*graph coloring*
symétries	*symmetries*
complexité	*complexity*

Bibliographie commentée

On appelle grille de sudoku de rang N un quadrillage carré de côté N^2, subdivisé en N^2 sous-carrés de côté N nommés régions. Le sudoku usuel est de rang 3. Une solution est une grille contenant en chacune de ses cases un entier entre 1 et N^2, et telle que chaque entier apparaît une et une seule fois par ligne, colonne, et région. On appelle sudoku une grille partiellement remplie qu'on cherche à compléter en solution.

Dans le cas du rang 3 (sudoku 9x9), il y a de l'ordre de 10^{21} solutions à la grille vide [3] et seulement de l'ordre de 10 solutions si on prend en compte les symétries [4]. Le nombre

asymptotique de solutions à la grille vide de rang N n'est pas précisément connu mais des bornes ont été proposées [6]. Cette étude donne aussi un résultat sur les polynômes chromatiques et utilise les sudokus pour l'illustrer. En effet, compléter un sudoku de rang N en solution revient à un problème de coloriage de graphe. Ces problèmes consistent à colorier chaque sommet d'un graphe de telle manière que deux sommets adjacents soient coloriés différemment. Ici on cherche à colorier N cases (les sommets) avec N^2 couleurs qui sont les entiers de 1 à N^2, et deux cases sont adjacentes si et seulement si elles appartiennent à la même ligne, colonne ou région. Très vite on en vient à étudier le nombre minimal de couleurs nécessaires pour colorier correctement un certain graphe : c'est le nombre chromatique. L'étude mathématique du rang 2 est plus facile : il n'y a que 2 solutions distinctes à la grille vide modulo symétries et on en déduit qu'il y a 288 solutions au total [6].

Pour tout $k>0$, déterminer s'il y a au moins k solutions à un sudoku de rang quelconque est un problème NP-complet [1]. En conséquence, il est difficile de programmer un solveur efficace pour les sudokus. La résolution des sudokus est aussi un cas particulier de problème de satisfaction de contraintes [2]. Classiquement, les solveurs utilisent du backtracking [5], une méthode par essais-erreurs pour explorer l'arbre des solutions partielles. Un intérêt de cette méthode est qu'on peut paramétrer le solveur par une question sur l'ensemble des solutions (son cardinal ou au moins deux de ses éléments par exemple) en explorant juste une partie de l'arbre des solutions.

Cependant, pour réduire le nombre d'essais-erreurs, il est essentiel d'avoir des mécanismes de propagation de contraintes permettant de compléter la grille par déduction sans modifier l' ensemble des solutions [2,5]. Dans le cas du sudoku de nombreuses techniques de propagation ont été développées pour la résolution manuelle [7,8]. Certaines de ces techniques manuelles peuvent être incorporées dans des solveurs automatiques. Elles consistent à propager des contraintes en s'appuyant sur la structure du graphe d'adjacence (dans le problème de coloriage mentionné cidessus). En fonction de la difficulté de ces techniques, et du nombre d'essais-erreurs nécessaires à la résolution, on peut évaluer de façon automatique la difficulté intrinsèque d'un sudoku donné pour un humain [9].

Problématique retenue

En théorie même une propagation simple permet de résoudre une certaine famille de sudoku en temps polynomial. Peut-on vraiment l'observer expérimentalement ? Quel effort de programmation cela demande t-il ? Comment utiliser les résultats mathématiques pour tester son programme ?

Objectifs du TIPE du candidat

Je me propose

- d'étudier des algorithmes de propagation, en particulier leur complexité, la classe des sudokus résolubles uniquement par propagation

- d'implémenter un solveur générique : sudokus de taille quelconque, traitement des solutions (nombre, unicité..), propagation (éventuellement) utilisée

- de valider expérimentalement mes implémentations, notamment sur les sudokus 4x4, 9x9 et 16x16 : les sudokus 4x4 étant bien connus et de petite taille, ils devraient être très commode pour mettre au point le programme

- de comparer expérimentalement l'effet de propagations sur l'efficacité

Références bibliographiques (ÉTAPE 1)

[1] T. YATO, T. SETA : Complexity and Completeness of Finding Another Solution and Its Application to Puzzles *: IEICE Transaction on Fundamentals of Electronics, Communications and Computer Sciences, Volume 86-A, pages 1052-1060, 2003*

[2] H. SIMONIS : Sudoku as a Constraint Problem *: In Fourth International Workshop on Modelling and Reformulating Constraint Satisfaction Problems (CP2005), 2005 Oct 1 (Vol. 12, pp. 13-27), https://ai.dmi.unibas.ch/_files/teaching/fs13/ki/material/ki10-sudoku-inference.pdf*

[3] F. JARVIS, B. FELGENHAUER : Mathematics of Sudoku I *: Mathematical Spectrum 39, p15-22, 2006*

[4] F. JARVIS, E. RUSSEL : Mathematics of Sudoku II *: Mathematical Spectrum 39, p54-58, 2006*

[5] P. VAN BEEK : Backtracking Search Algorithms *: Handbook of Constraint Programming, Elsevier, chapitre 4, p83-122, 2006*

[6] A.M. HERZBERG, M. RAM MURTY : Sudoku Squares and Chromatic Polynomials *: Notice of the AMS, volume 54(6), p708-717, 2007*

[7] J.F. CROOK : A Pencil-and-Paper Algorithm for Solving Sudoku Puzzles *: Notice of the AMS, volume 56(4), 2009*

[8] T. DAVIS : The Mathematics of Sudoku *: http://www.geometer.org/mathcircles/sudoku.pdf, 2012*

[9] R. PELÁNEK : Difficulty Rating of Sudoku Puzzles : An Overview and Evaluation *: ArXiv, https://doi.org/10.48550/arXiv.1403.7373, 2014*