Projet : résolution de sudokus par programmation linéaire

Un sudoku est un problème qui consiste à ranger dans une grille de 9 cases sur 9 des entiers entre 1 et 9, en respectant les contraintes suivantes :

1. chaque ligne de la grille contient une et une seule fois chaque entier entre 1 et 9,
2. chaque colonne de la grille contient une et une seule fois chaque entier entre 1 et 9,
3. chaque sous-carré de la forme $(3i+1, 3i+3) \times (3j+1, 3j+3)$, $0 \leq i, j \leq 2$ contient une et une seule fois chaque entier entre 1 et 9.

Certaines cases de la grille sont préremplies, il s'agit alors de trouver une façon de compléter la grille. Le but du projet est d'étudier et mettre en œuvre un algorithme de résolution de sudokus.

Remarque : la dimension importante du problème pouvant rendre la convergence des algorithmes difficile, on pourra commencer par des sudokus 4×4 pour alléger les calculs.

1. Formalisation du problème

On considère, pour $i, j, k \in \{1, \cdots, 9\}$, la variable x_{ijk} définie par

$$x_{ijk} = \begin{cases} 1, & \text{si la case } (i,j) \text{ contient le chiffre } k, \\ 0, & \text{sinon.} \end{cases}$$

Ainsi, une solution du sudoku est entièrement déterminée par la donnée d'un vecteur $x \in X = \{0,1\}^{729}$.

(a) Traduire les contraintes 1, 2 et 3 par des contraintes sur le vecteur x, de la forme $g_l(x) = 0, 1 \leq l \leq M$. On explicitera également le fait qu'une case donnée (i,j) ne contient qu'un chiffre. On note X_0 l'ensemble des $x \in X$ satisfaisant ces contraintes. (On n'impose pas pour le moment la contrainte liée aux cases préremplies.)

(b) On considère pour une grille donnée l'ensemble $T \subset \{1, \cdots 9\}^2$ des cases préremplies, et la fonction

$$f : T \to \{1, \cdots 9\}$$

donnant les valeurs de ces cases. On définit alors un vecteur $y \in X$ par

$$y_{ijk} = \begin{cases} 1, & \text{si } (i,j) \in T \text{ et } f(i,j) = k, \\ 0, & \text{sinon.} \end{cases}$$

Définir à l'aide du vecteur y une fonction linéaire $J : [0,1]^{729} \to \mathbb{R}$ telle que les solutions du sudoku correspondent aux points de minimum de J sur X_0. On peut donc écrire le problème

$$\text{Trouver } x = (x_{ijk})_{1 \leq i,j,k \leq 9} \in X_0 \text{ tel que } J(x) = \inf_{y \in X_0} J(y). \tag{1}$$

2. Relaxation linéaire

Pour appliquer l'algorithme du simplexe, il faut se ramener à un probème linéaire, ce qui impose de se débarrasser de la contrainte d'intégrité des x_{ijk}. On remplace donc la contrainte $x_{ijk} \in \{0,1\}$ par $x_{ijk} \in [0,1]$, ce qui conduit à

$$\text{Trouver } x = (x_{ijk})_{1\leq i,j,k\leq 9} \in Y_0 \quad \text{ tel que } J(x) = \inf_{y\in Y_0} J(y). \tag{2}$$

où

$$Y_0 = \{x \in [0,1]^{729}, g_l(x) = 0, 1 \leq l \leq M\}.$$

On peut alors résoudre le problème (2) par l'algorithme du simplexe, par exemple. Cependant, la solution trouvée ne sera a priori pas entière. On introduit donc le schéma suivant pour obtenir des solutions entières.

– On résout la relaxation linéaire (2). Si, par chance, la solution est entière, on a fini. Sinon, on répète le schéma itératif suivant.

– On commence par trier les 9^3 variables par distance décroissante à $\mathbb{N}$, i.e. selon la valeur de $|x_{ijk} - 0.5|$. On considère maintenant les variables dans cet ordre. Soit x_{ijk} la variable courante ; si fixer x_{ijk} à 1 rend le programme linéaire (2) irréalisable (i.e. si l'ensemble des points admissibles est vide), alors on le fixe à 0 (i.e. on rajoute la contrainte $x_{ijk} = 0$), on résout à nouveau la relaxation linéaire (2), et on recommence. Dans le cas contraire, si fixer x_{ijk} à 0 rend le programme linéaire (2) irréalisable, on le fixe à 1, on résout à nouveau la relaxation linéaire, et on recommence. Enfin, si aucun des choix 0 ou 1 ne rend le programme linéaire (2) irréalisable, on passe à la variable suivante.

– On arrête ce programme itératif soit lorsqu'une solution entière est trouvée, soit lorsqu'on n'a pas pu fixer de variable supplémentaire.

Suggestions pour l'étude du problème

Les suggestions ci-dessous ne sont que des pistes de réflexion, il n'est pas obligatoire de traiter toutes les questions, et d'autres aspects pourront être abordés. Cependant, le travail comportera obligatoirement une partie de programmation.

(a) Justifier qu'on peut initialiser l'algorithme du simplexe par la grille triviale

$$\begin{pmatrix} 1 & 2 & 3 & \ldots & 9 \\ 2 & 3 & 4 & \ldots & 1 \\ 3 & 4 & 5 & \ldots & 2 \\ \vdots & \vdots & \vdots & & \vdots \\ 9 & 1 & 2 & \ldots & 8 \end{pmatrix}.$$

(b) Programmer la méthode de résolution proposée en utilisant l'algorithme du simplexe pour résoudre la relaxation linéaire (2).

(c) Tester cette méthode sur des exemples. Commenter la convergence vers une solution entière et les temps de calcul.

(d) Comparer éventuellement avec d'autres types de méthodes (combinatoires par exemple).