

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE
UNIVERSITE AKLI MOAND OULHADJE-BOUIRA



Faculté des Sciences et des Sciences Appliquées
Département de Mathématiques

Mémoire de fin d'études

Présenté par :

Metidji Nour Elislam

En vue de l'obtention du diplôme de **Master** en :

Filière : MATHÉMATIQUES

Spécialité : Recherche Opérationnelle

Thème :

Modélisations du jeu SuDoku³

Devant le jury composé de :

M. BEDDEK Said	MCB	UAMOB	Président
M. KHELIFA Yousfi	MAA	UAMOB	Encadreur
M. DEMMOUCHE Nacer	MCB	UAMOB	Examinateur
Mme ALEM Lala Maghnia	MAA	UAMOB	Examinatrice

Année Universitaire 2018/2019

Table des matières

Remerciements	iv
Résumé	v
Abstract	vi
Introduction	vii
1 LE JEU SuDoku ET SON EVOLUTION	1
1.1 Historique du Jeu SuDoku	1
1.2 Présentation du jeu SuDoku ³	3
1.2.1 Support du jeu	3
1.2.2 Les régions	3
1.2.3 Les Règles du jeu	12
1.3 Particularités du SuDoku ³	12
1.3.1 Caractéristiques d'une grille de SuDoku ^k	12
1.3.2 Dénombrement les possibilités des solutions	13
1.3.3 Equivalences de grille SuDoku ³	14
1.3.4 Existence de la solution	15
1.3.5 Unicité de la solution	17
2 PROBLEMES DE SATISFACTION DE CONTRAINTES ET DE COLORATION DANS UN GRAPHE	19
2.1 Le Problème de Satisfaction de Contraintes (CSP)	19
2.1.1 Programmation linéaire	19
2.1.2 Problème de satisfaction des contraintes (CSP)	20
2.2 Quelques Rappels sur théorie des graphes	22

3	CHEMINEMENT DU JEU SuDoku³ AU PROBLEME DE SATISFACTION DE CONTRAINTES	24
3.1	Présentation du Problème de Satisfaction de Contraintes . . .	24
3.1.1	Variables dans les modèles SuDoku ³	24
3.1.2	contraintes dans les modèles SuDoku	25
3.1.3	Énoncé du programme linéaire en nombres entiers . . .	28
3.2	Cheminement de SuDoku ³ au CSP	28
3.3	Exemple (<i>SuDoku</i> ² \rightarrow <i>CSP</i>)	29
3.3.1	Étape 0 :	30
3.3.2	Étape 1 :	30
3.3.3	Étape 2 :	31
3.3.4	Étape 3 :	32
3.3.5	Étape 4 :	33
3.3.6	Étape 5 :	34
4	CHEMINEMENT DU JEU SuDoku³ AU PROBLEME DE COLORATION DE SOMMETS	36
4.1	Introduction	36
4.2	Cheminement de SuDoku ³ à la Coloration de Sommets dans un graphe : $G = (Y, E)$	37
4.2.1	Les étapes de coloration	37
4.3	Exemple (SuDoku ² \rightarrow Coloration dans $G = (Y, E)$)	38
Conclusion générale		46
Annexe		52

Remerciements

Tout d'abord, je tiens à remercier le bon Dieu le tout Puissant de m'avoir donné la force et le courage de mener à bien ce modeste travail, également je remercie infiniment mes parents, qui m'ont encouragé et aidé à arriver à ce stade de ma formation.

Je tiens à remercier tous ceux et celle qui ont contribué à finaliser ce modeste travail.

Mes remerciements vont à monsieur **Khelifa Yousfi** mon encadreur pour m'avoir guidé pour la réalisation de ce projet.

Je remercie vivement **les étudiants** Master Recherche Opérationnelle pour leur aide morale durant toute la période de préparation.

Résumé

Le jeu SuDoku, dont les premières versions remontent à environs mille ans, demeure une occupation ludique de tous les âges. Il serait intéressant d'en faire quelques approches de modélisations à l'aide de modèles mathématiques, identifiés, connus utilisant des concepts théoriques de recherche opérationnelle tels que : la théorie des graphes, le problème de Programmation Linéaire ou le Problème de Satisfaction de Contraintes. . .

Après un rappel historique sur les origines du jeu, son évolution, sa complexité, il s'agit de s'intéresser, au jeu SuDoku défini par une grille $3^2 \times 3^2$ et qui sera noté SuDoku³. Il faut, notamment, en donner une représentation pratique complète, énumérer les règles qui l'animent, évaluer ses difficultés pratiques de résolution. L'ensemble du vocabulaire utilisé dans le cadre du jeu SuDoku³ doit être identifié et répertorié.

Il faut expliquer, de façon didactique, pour chaque modèle équivalent identifié et retenu, le cheminement séparant la grille SuDoku³ à résoudre, du modèle personnalisé et conforme aux paramètres de la « grille-mère ». Pour ce faire, une illustration appropriée, pourrait aider à la compréhension. Peut-on, pour n entier, généraliser les résultats éventuels à une grille SuDoku ^{n} ?

Mots clés :

SuDoku, Modélisation, Problème de Satisfaction de Contraintes, Coloration dans un graphe.

Abstract

The SuDoku game, the first versions of which date back around a thousand years, remains a fun activity for all ages. It would be interesting to make some modeling approaches using mathematical models, identified, known using theoretical concepts of operational research such as : graph theory, the problem of Linear Programming or the Problem of Satisfaction of Constraints. . .

After a historical reminder on the origins of the game, its evolution, its complexity, it is a question of being interested, in the game SuDoku defined by a grid $3^2 \times 3^2$ and which will be noted SuDoku³. It is necessary, in particular, to give a complete practical representation of it, to list the rules which animate it, to evaluate its practical difficulties of resolution. All of the vocabulary used in the SuDoku³ game must be identified and listed.

It is necessary to explain, in a didactic way, for each equivalent model identified and retained, the path separating the SuDoku³ grid to be solved, from the personalized model and in accordance with the parameters of the "mother grid". To do this, an appropriate illustration could help understanding. Can we, for n integer, generalize the possible results to a SuDoku ^{n} grid?

Key-Words :

SuDoku, Modeling, Constraints Satisfaction Problem, Coloring in a graph.

Introduction

La Recherche Opérationnelle propose un ensemble de méthodes mathématiques pour résoudre des problèmes d'optimisation liés aux organisations du monde réel : problèmes de logistique, d'emploi du temps, de gestion des flux, de transport...etc.

Le jeu est une activité de tout être vivant, en ce sens qu'il met en scène une situation et permet donc de s'y plonger, de la vivre concrètement et, si on le désire, y réfléchir. On pourrait s'attendre à ce qu'un jeu nécessitant de la logique ne fasse appel qu'à un très petit nombre de personnes - peut-être des mathématiciens, des informaticiens ou des joueurs professionnels. Cependant, en très peu de temps, SuDoku a acquis une popularité exceptionnelle,

Des jeux mathématiques préliminaires sont introduits et quelques exemples des jeux les plus connus sont présentés. Dans ce memoire, un jeu mathématique très populaire, appelé «SuDoku»(chiffre-unique), est analysé dans tous les détails.

Ce jeu consiste à remplir la grille avec des chiffres, des symboles, ou des couleurs. Ceci doit se faire en respectant des règles.

Au départ, cette grille de carré possède quelque case déjà remplies, et le joueur doit remplir le reste en respectant ses règle, et chaque remplissage, il ya une case qui n'admet q'une seule valeur

La complexité mathématique du jeu SuDoku a été étudiée pour la comparer similaire. L'unicité de la solution de SuDoku peut être posé.

on considère qu'un problème de SuDoku est bien posé s'il admet exactement une solution unique.

Un jeu bien conçu ne devraient pas contenir d'information redondante.

Le jeu SuDoku soulève des questions :

Combien de grilles SuDoku peuvent être générées pour une dimension donnée ? Quel est le plus petit nombre de numéros de départ permettant une

solution unique ? Quelle niveau de complexité possède la modélisation du jeu SuDoku ?.

La modélisation du jeu SuDoku est compliquée. La plus grande contribution de cette memoir est peut-être de fournir tous les détails de la modélisation du SuDoku afin que la carte des différences puisse le résoudre, c'est très ingénieux.

Après modélisation propre du jeu SuDoku³, nous essayerons de montrer son équivalence avec deux autres modèles connus en recherche opérationnelles à savoir d'une part :

Le problème de satisfaction de contraintes noté CSP, d'autre part le problème de Coloration de sommets dans un graphe.

Notre travail se déroulera en quatre chapitres :

Dans le premier Chapitre, nous parlerons sur le jeu SuDoku, où nous présentons son histoire et son évolution, en plus des particularités de ce jeu.

Dexième chapitre : dans ce chapitre, nous rappelons quelques notions sur le problème de satisfaction de contraintes (CSP) et Coloration de sommets dans un graphe.

Troisième chapitre : nous proposons un cheminement par étapes du modèle propre au jeu SuDoku³ au problème de satisfaction de contraintes.

Quatrième chapitre : ici, le cheminement se fera, toujours par étapes du modèle propre au jeu SuDoku³ au Coloration de sommets dans un graphe.

Deux exemples pratiques illustreront chacune des parties.

Enfin nous terminerons par une conclusion et des perspectives sous forme des questions.

Chapitre 1

LE JEU SuDoku ET SON EVOLUTION

1.1 Historique du Jeu SuDoku

Le mot SuDoku signifie nombre (Su) unique (Doku) en Japonais [1].

Le jeu SuDoku est un jeu de réflexion très ancien mais au succès récent. En effet, ses origines viennent du **carrés magiques**⁽⁴⁾ qui existe en Chine depuis des milliers d'années, mais aussi du **carrés latins**⁽⁵⁾ du mathématicien suisse **Leonhard Euler**⁽³⁾ qui imagine un problème sous la forme d'une grille avec des chiffres, des couleurs ou toute autres éléments, puisque les chiffres ne sont qu'une simple composante et non obligatoire pour la résolution du jeu.

C'est en 1979 [1], qu'un pigiste spécialisé dans les puzzles, **Howard Garns**⁽²⁾, crée le premier jeu tel que nous le connaissons aujourd'hui. Dell Magazine est le premier à publier ses grilles et a utilisé des chiffres dans ses publications. Par contre, Scramblets, de Penny Press, et SuDoku Word, de Knight Features Syndicate, utilisent tous les deux des lettres.

L'intérêt du jeu réside dans la simplicité de ses règles, et dans la complexité de ses solutions. Les grilles publiées ont souvent un niveau de difficulté indicatif. En général, les grilles contenant le plus de chiffres préremplis fussent les plus simples, l'inverse n'est pas systématiquement vrai. La véritable difficulté du jeu réside plutôt dans la difficulté à trouver la suite exacte de chiffres à ajouter.

Ce jeu a déjà inspiré plusieurs versions de SuDoku électroniques qui ap-

portent un intérêt différent à la résolution des grilles de SuDoku. Sa forme en grille et son utilisation ludique le rapprochent d'autres casse-tête publiés dans les journaux, tels les mots croisés et les problèmes d'échecs, le SuDoku est partout.

Des professeurs recommandent la pratique du SuDoku comme un entraînement aux raisonnements logiques. Le niveau de difficulté peut dans ce cas être adapté au public.

En 1986, **Nikoli**⁽¹⁾ [1] introduit deux nouveautés, qui rendront le jeu de SuDoku populaire : le nombre de dévoilés est au plus de 30 et les grilles sont symétriques, c'est-à-dire que les dévoilés sont symétriquement distribués autour du centre de la grille. Aujourd'hui, la plupart des journaux importants au Japon, tel Asahi Shimbun, publient ce jeu. Au Japon, Nikoli est toujours propriétaire du nom SuDoku, ses concurrents utilisent un autre nom.

C'est en juillet 2005 que le SuDoku arrive en France, publié par Sport cérébral, éditeur spécialisé dans les jeux de réflexion. Le premier numéro se vendra à 20 000 exemplaires soit deux fois plus qu'à l'accoutumée lors de la sortie d'un nouveau jeu, un record selon Xavier de Bure, directeur général de l'éditeur. Le Figaro publie les premières grilles quotidiennes dès le début juillet, suivi au cours de l'été 2005 par Libération, La Provence, Nice Matin, 20 Minutes, Métro et Le Monde. Le magazine 1, 2, 3... SuDoku sortit son premier numéro en novembre 2005.

Le jeu SuDoku connaît une énorme popularité médiatique dans le monde, que ce soit en France, en Angleterre ou au Etats-Unis, le succès est impressionnant et ne cesse de grandir. Les grilles de SuDoku sont incroyablement captivante!

Cette brusque augmentation de popularité dans les journaux britanniques et internationaux fait que le SuDoku est considéré comme le « cube de Rubik du XXIe siècle » (traduction libre de « the Rubik's cube of the 21st century »). À titre d'exemple, Wayne Gould fournit fin 2005 des grilles pour environ 70 quotidiens dans 27 pays.

(1); (2); (3); (4); (5) voir l'annexe.

1.2 Présentation du jeu SuDoku³

1.2.1 Support du jeu

Définition 1.2.1 Une Case est un objet de forme carrée qui peut contenir un nombre entier, ce nombre appartient à $I_n = \{1, 2, \dots, n\}$, si la case est vide, au début du jeu, elle est appelée alors **Given**, sinon la case est vide, dans toute la suite la case sera considérée comme unité, notée "u" de mesure des éléments du jeu SuDoku.

Remarque 1.2.1 dans notre travail $n = 9$.

Définition 1.2.2 La grille du jeu SuDoku³ est un Tableau carré de $9u \times 9u$ est composé de 9 sous-tableaux appelés **bloc** carré de dimension $3u \times 3u$.

La grille est notée $C = (c_{ij})_{i,j \in I_9}$ matrice carrée.

Définition 1.2.3 On appelle **ordre** d'une grille de SuDoku^k, l'ordre de la matrice $C = (c_{ij})_{i,j \in I_n}$, avec $n = k^2$, et k représente la taille du bloc.

Définition 1.2.4 Le jeu SuDoku³ consiste à compléter les cases vides de la grille, à l'aide de chiffres entre 1 et 9, en évitant la répétition d'un même chiffre dans une même région ligne, colonne, ou bloc.

Définition 1.2.5 Une **bande** est une suite de blocs adjacents sur l'axe horizontal. Une **pilier** est une suite de blocs adjacents sur l'axe vertical.

Dans un SuDoku³, il y a ainsi 3 bandes et 3 pilier.

Définition 1.2.6 On notera $V = (v_{ij})_{i,j \in I_9}$, la matrice contenant les valeurs des cases de la grille.

1.2.2 Les régions

Dénomination des régions

Toute grille de SuDoKu³ comporte :

1. 9 régions lignes $(RL_i)_{i \in I_9}$ qui représentent les lignes de la grille.
2. 9 régions colonnes $(RC_j)_{j \in I_9}$ qui représentent les colonnes de la grille.
3. 9 régions carrés $(RK_{i_0,j_0})_{i_0,j_0 \in I_3}$ qui représentent les carrés de la grille.

région ligne (RL_i)

avec $i = 3$

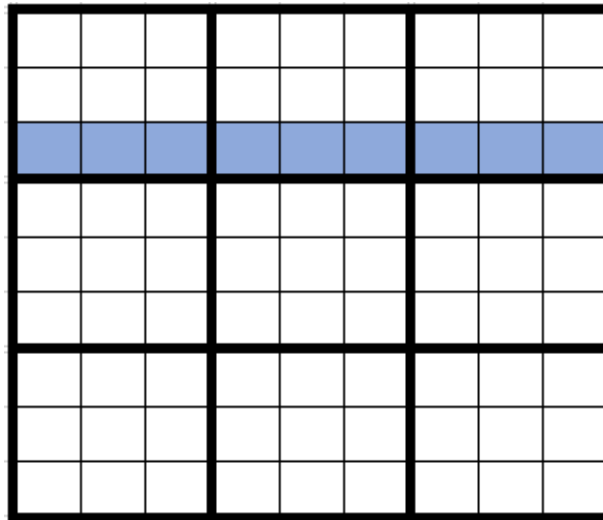


fig.1.1 : région ligne

région colonne (RC_j)

avec $j = 8$

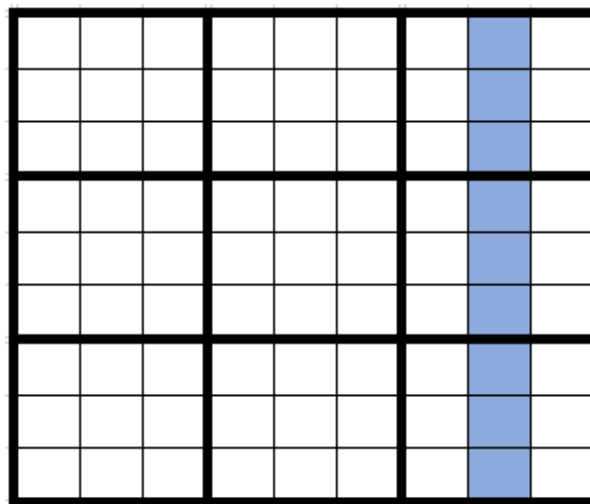


fig1.2 : région colonne

région carré ($RK_{i_0 j_0}$)
avec $(i_0, j_0) = (2, 1)$

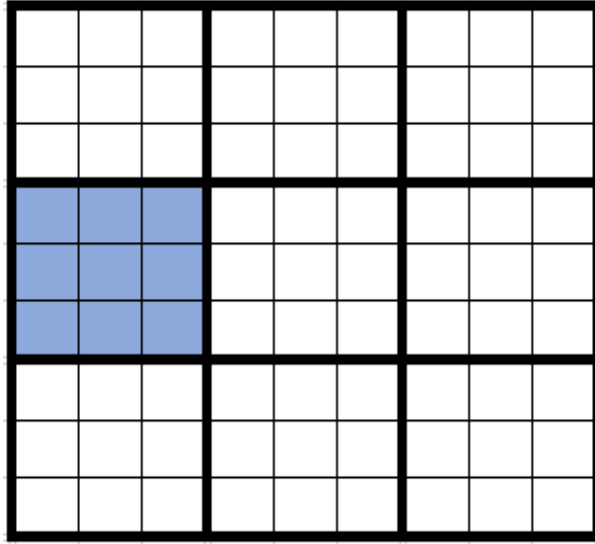


fig.1.3 région carré

Propriétés des Régions

1) Intersection des région de même type :

$$\forall (i_1, i_2) \in I_9^2 : i_1 \neq i_2 \implies RL_{i_1} \cap RL_{i_2} = \emptyset$$

$$\forall (j_1, j_2) \in I_9^2 : j_1 \neq j_2 \implies RC_{j_1} \cap RC_{j_2} = \emptyset$$

$$\forall (i_1, j_1, i_2, j_2) \in I_3^4 : (i_1, j_1) \neq (i_2, j_2) \implies RK_{i_1, j_1} \cap RK_{i_2, j_2} = \emptyset$$

2) Réunion des régions de même type :

$$\bigcup_{i=1}^9 RL_i = \bigcup_{j=1}^9 RC_j = \bigcup_{i_0=1}^3 \bigcup_{j_0=1}^3 RK_{i_0, j_0} = C$$

les parties **1)** et **2)** montrant d'un même type (ligne, colonne, ou bloc carré) forment une partition de C

3) Intersection des régions de type différent :

il s'agit d'intersections en couples ou triplets :

(RC, RL) ;

(RK, RC) ;

(RK, RL) ;

(RK, RL, RC) .

a) cas de l'intersection région ligne, région colonne :

$$\forall (i, j) \in I_9^2 : RL_i \cap RC_j = c_{ij}$$

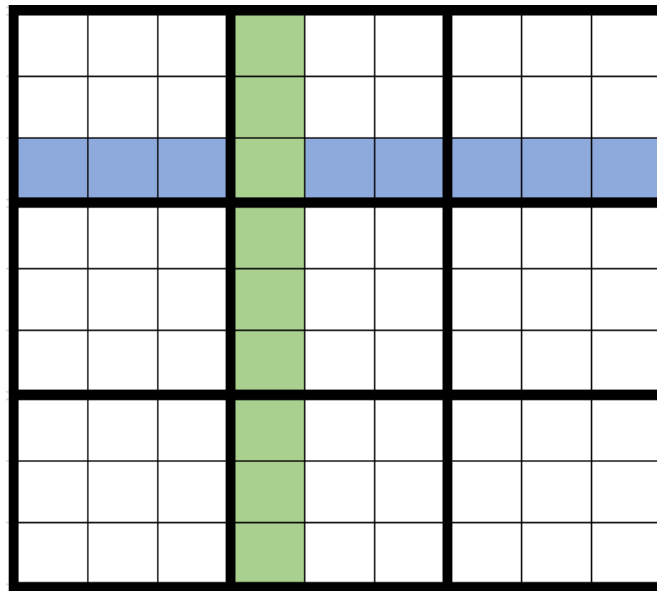


fig 1.4 $RL_i \cap RC_j$

b) cas de l'intersection région ligne, région carrée :

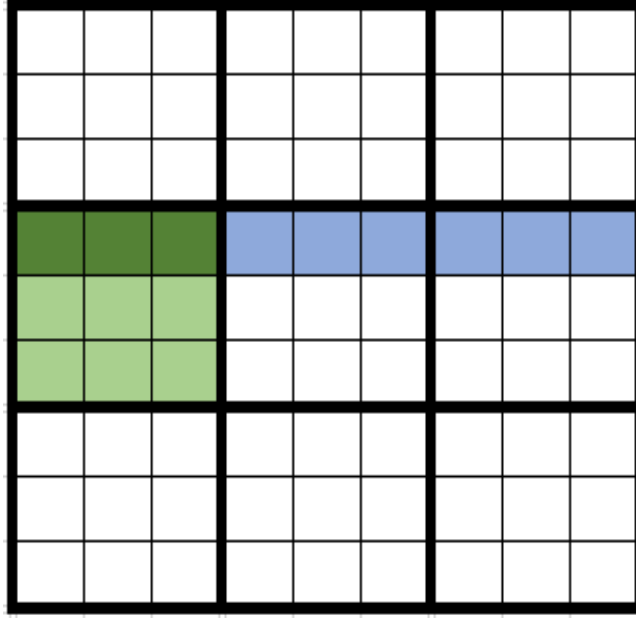


Fig 1.5 $RL_i \cap RK_{i_0j_0}$

pour les régions carrées : $(i_0, j_0) \in I_3^2$:

Notation :

$$\begin{aligned} \varphi : I_3 \times I_3 &\rightarrow I_9 \\ (s_0, s) &\rightarrow \varphi(s_0, s) = 3(s_0 - 1) + s \end{aligned}$$

$$RK_{i_0j_0} = (c_{\varphi(i_0,i),\varphi(j_0,j)})_{(i,j) \in I_3^2}$$

$$RK_{i_0j_0} = \begin{pmatrix} c_{\varphi(i_0,1),\varphi(j_0,1)} & c_{\varphi(i_0,1),\varphi(j_0,2)} & c_{\varphi(i_0,1),\varphi(j_0,3)} \\ c_{\varphi(i_0,2),\varphi(j_0,1)} & c_{\varphi(i_0,2),\varphi(j_0,2)} & c_{\varphi(i_0,2),\varphi(j_0,3)} \\ c_{\varphi(i_0,3),\varphi(j_0,1)} & c_{\varphi(i_0,3),\varphi(j_0,2)} & c_{\varphi(i_0,3),\varphi(j_0,3)} \end{pmatrix}$$

$$RK_{i_0j_0} = \begin{pmatrix} c_{3i_0-2,3j_0-2} & c_{3i_0-2,3j_0-1} & c_{3i_0-2,3j_0} \\ c_{3i_0-1,3j_0-2} & c_{3i_0-1,3j_0-1} & c_{3i_0-1,3j_0} \\ c_{3i_0,3j_0-2} & c_{3i_0,3j_0-1} & c_{3i_0,3j_0} \end{pmatrix} \quad (1.1)$$

Voici la représentation en grille des valeurs $v_{\varphi(i_0,i),\varphi(j_0,j)}$ telle que $(i_0, j_0, i, j) \in I_3^4$

$\varphi(j_0,j)$		$j_0 = 1$			$j_0 = 2$			$j_0 = 3$		
		$j=1$	$j=2$	$j=3$	$j=1$	$j=2$	$j=3$	$j=1$	$j=2$	$j=3$
$\varphi(i_0,i)$	$i=1$	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}	v_{16}	v_{17}	v_{18}	v_{19}
	$i=2$	v_{21}	v_{22}	v_{23}	v_{24}	v_{25}	v_{26}	v_{27}	v_{28}	v_{29}
	$i=3$	v_{31}	v_{32}	v_{33}	v_{34}	v_{35}	v_{36}	v_{37}	v_{38}	v_{39}
$i_0 = 2$	$i=1$	v_{41}	v_{42}	v_{43}	v_{44}	v_{45}	v_{46}	v_{47}	v_{48}	v_{49}
	$i=2$	v_{51}	v_{52}	v_{53}	v_{54}	v_{55}	v_{56}	v_{57}	v_{58}	v_{59}
	$i=3$	v_{61}	v_{62}	v_{63}	v_{64}	v_{65}	v_{66}	v_{67}	v_{68}	v_{69}
$i_0 = 3$	$i=1$	v_{71}	v_{72}	v_{73}	v_{74}	v_{75}	v_{76}	v_{77}	v_{78}	v_{79}
	$i=2$	v_{81}	v_{82}	v_{83}	v_{84}	v_{85}	v_{86}	v_{87}	v_{88}	v_{89}
	$i=3$	v_{91}	v_{92}	v_{93}	v_{94}	v_{95}	v_{96}	v_{97}	v_{98}	v_{99}

Fig.1.6 : représentation en grille

Localisation d'un indice i_1 de RL_{i_1} au coordonnées (i_0, j_0) de $RK_{(i_0, j_0)}$:

Il s'agit de positionner l'indice de RL_{i_1} par rapport à la région carrée

$RK_{(i_0, j_0)}$

On a :

$$i_1 \in \bigcup_{i=1}^3 \{\varphi(i_0, i)\} \iff i_1 \in \{3(i_0 - 1) + 1, 3(i_0 - 1) + 2, 3(i_0 - 1) + 3\}$$

$$i_1 \in \bigcup_{i=1}^3 \{\varphi(i_0, i)\} \iff i_1 \in \{3i_0 - 2, 3i_0 - 1, 3i_0\}; \forall (i_0, j_0) \in I_3^2 \quad (1.2)$$

$$RK_{i_0, j_0} \cap RL_{i_1} = \left(\bigcup_{j=1}^3 c_{i_1, \varphi(j_0, j)} \right) = (c_{i_1, 3(j_0-1)+1}, c_{i_1, 3(j_0-1)+2}, c_{i_1, 3(j_0-1)+3})$$

$$RK_{i_0, j_0} \cap RL_{i_1} = (c_{i_1, 3j_0-2}, c_{i_1, 3j_0-1}, c_{i_1, 3j_0}) \quad (1.3)$$

c) Cas de l'intersection région colonne, région carrée :

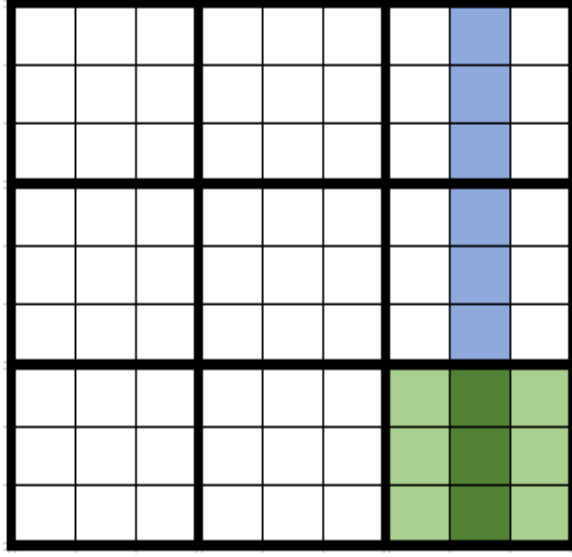


Fig 1.6 $RK_{i_0, j_0} \cap RC_j$

Exactement le même raisonnement qu'il suffit de remplacer RL_{i_1} par RC_{j_1}

Localisation d'un indice colonne j_1 de RC_{j_1} au coordonnées (i_0, j_0)

de $RK_{(i_0, j_0)}$:

Il s'agit de positionner l'indice de RL_{i_1} par rapport à la région

$RK_{(i_0, j_0)}$

$$j_1 \in \bigcup_{j=1}^3 \{\varphi(j_0, j)\} \iff j_1 \in \{3(j_0 - 1) + 1, 3(j_0 - 1) + 2, 3(j_0 - 1) + 3\}$$

$$j_1 \in \bigcup_{j=1}^3 \{\varphi(j_0, j)\} \iff j_1 \in \{3j_0 - 2, 3j_0 - 1, 3j_0\}; \forall (i_0, j_0) \in I_3^2 \quad (1.4)$$

$$RK_{i_0, j_0} \cap RC_{j_1} = \left(\bigcup_{i=1}^3 c_{\varphi(i_0, i), j_1} \right) = (c_{3(i_0-1)+1, j_1}, c_{3(i_0-1)+2, j_1}, c_{3(i_0-1)+3, j_1})$$

$$RK_{i_0, j_0} \cap RC_{j_1} = (c_{3i_0-2, j_1}, c_{3i_0-1, j_1}, c_{3i_0, j_1}) \quad (1.5)$$

Exemple 1.2.1

$$i_0 = 3; j_0 = 2; (1.2) \text{ et } (1.4) \implies \begin{cases} i_1 \in \{3 \times 3 - 2, 3 \times 3 - 1, 3 \times 3\} = \{7, 8, 9\} \\ j_1 \in \{3 \times 2 - 2, 3 \times 2 - 1, 3 \times 2\} = \{4, 5, 6\} \end{cases}$$

En utilisant (1.1)

$$RK_{32} \cap RL_{i_1} = \begin{pmatrix} c_{74} & c_{75} & c_{76} \\ c_{84} & c_{85} & c_{86} \\ c_{94} & c_{95} & c_{96} \end{pmatrix} \cap RL_{i_1}$$

En utilisant (1.3)

$$RK_{32} \cap RL_{i_1} = (c_{i_1, 3 \times 2 - 2}, c_{i_1, 3 \times 2 - 1}, c_{i_1, 3 \times 2}) = (c_{i_1, 4}, c_{i_1, 5}, c_{i_1, 6})$$

En utilisant (1.1)

$$RK_{32} \cap RC_{j_1} = \begin{pmatrix} c_{74} & c_{75} & c_{76} \\ c_{84} & c_{85} & c_{86} \\ c_{94} & c_{95} & c_{96} \end{pmatrix} \cap RC_{j_1}$$

En utilisant (1.5)

$$RK_{32} \cap RC_{j_1} = (c_{3 \times 3 - 2, j_1}, c_{3 \times 3 - 1, j_1}, c_{3 \times 3, j_1}) = (c_{7, j_1}, c_{8, j_1}, c_{9, j_1})$$

d) L'intersection de 3 types de région déferant :

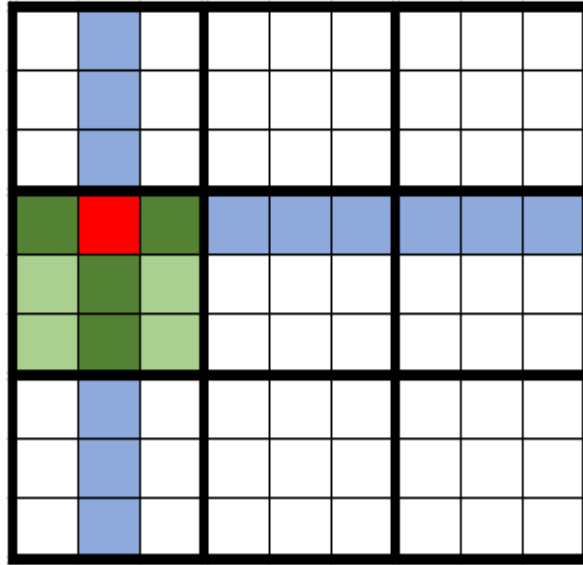


Fig 1.7 $RK_{i_0, j_0} \cap RL_{i_1} \cap RC_{j_1}$

Soit $\forall (i_1, j_1) \in I_9^2$, on a $(c_{i_1 j_1}) = RL_{i_1} \cap RC_{j_1}$

Question :

$\exists ? (i_0, j_0) \in I_9^2 / (c_{i_1 j_1}) \cap RK_{i_0, j_0} = (c_{i_1 j_1})$

Réponse :

$$RK_{i_0, j_0} \cap RL_{i_1} \cap RC_{j_1} = \begin{cases} (c_{i_1 j_1}) & \text{si } (c_{i_1 j_1}) \in RK_{i_0, j_0} \\ \emptyset & \text{sinon} \end{cases}$$

Remarque :

$$|RK_{i_0, j_0} \cap RL_{i_1} \cap RC_{j_1}| = \begin{cases} 1 & \text{si } (c_{i_1 j_1}) \in RK_{i_0, j_0} \\ 0 & \text{sinon} \end{cases}$$

1.2.3 Les Règles du jeu

Dans ce jeu il existe une règle qui s'applique à l'ensemble des région, c'est à dire les 9 lignes et les 9 colonnes et les 9 blocs

Région colonne $(RC_j)_{j \in I_9}$: les valeurs sont différentes deux à deux :

$$\forall i_1, i_2 \in I_9 \quad i_1 \neq i_2 \iff v_{i_1 j} \neq v_{i_2 j}$$

Régions lignes $(RL)_{i \in I_9}$: les valeurs sont différentes deux à deux :

$$\forall j_1, j_2 \in I_9 \quad j_1 \neq j_2 \iff v_{i j_1} \neq v_{i j_2}$$

Régions carrées $(RK)_{i_0, j_0 \in I_3}$: les valeurs sont différentes deux à deux :

$$\forall (i_1, i_2, j_1, j_2) \in I_3^4$$

$$\begin{cases} v_{i_1, j_1} \in RK_{i_0, j_0} \\ v_{i_2, j_2} \in RK_{i_0, j_0} \\ (i_1, j_1) \neq (i_2, j_2) \end{cases} \implies v_{i_1, j_1} \neq v_{i_2, j_2}$$

1.3 Particularités du SuDoku³

1.3.1 Caractéristiques d'une grille de SuDoku^k

Pour Imaginer profondément une structure du grille C de SuDoku^k de dimension $k^2 \times k^2$, on peut déduire les résultat suivant :

Toute grille de SuDoku^k comporte :

k^2 régions lignes $(RL_i)_{i \in I_{k^2}}$ qui représentent les lignes de la grille.

k^2 régions colonnes $(RC_j)_{j \in I_{k^2}}$ qui représentent les colonnes de la grille.

k^2 régions carrés $(RK_{i_0, j_0})_{i_0, j_0 \in I_k}$ qui représentent les carrés de la grille.

chaque régions ligne comporte k^2 cases

chaque régions colonne comporte k^2 cases

chaque régions carrée comporte k^2 cases

Donc toute grille de SuDoku^k comporte k^4 cases.

Toute grille de SuDoku^k comporte k bandes $(B_{i_0})_{i_0 \in I_k}$ et k piliers $(P_{j_0})_{j_0 \in I_k}$.

chaque bande comporte k ligne

chaque pilier comporte k colonne

$$B_{i_0} = \bigcup_{j_0=1}^k RK_{i_0, j_0}, \quad i_0 \in I_k$$

$$P_{j_0} = \bigcup_{i_0=1}^k RK_{i_0, j_0}, \quad j_0 \in I_k$$

Nous résumons tout cela dans la relation suivante :

$$C = \bigcup_{i_0=1}^k B_{i_0*} = \bigcup_{j_0=1}^k P_{*j_0} = \bigcup_{i_0=1}^k \bigcup_{j_0=1}^k RK_{i_0j_0} = \bigcup_{i=1}^{k^2} RL_i = \bigcup_{j=1}^{k^2} RC_j = \bigcup_{i=1}^{k^2} \bigcup_{j=1}^{k^2} (c_{ij})$$

Problème :

Soit un jeu de SuDoku^k. Soit $N(k)$ le plus grande nombre de cases de la grille de ce jeu qui sont dépendantes de la case c_{ij} avec $(i, j) \in I_n^2$ ou $n = k^2$.

Trouver $N(k)$ pour $k \in \mathbb{N}^*$

Solution :

Nous avons déjà vu que chaque case d'une grille de SuDoku^k dépend de $(k^2 - 1)$ cases du même bloc et de $(k^2 - k)$ cases de la même colonne et $(k^2 - k)$ cases de la même ligne.

La sommation de ses trois résultats donne :

$$N(k) = 2(k^2 - k) + k^2 - 1$$

$$N(k) = 3k^2 - 2k - 1$$

Démonstration :

Nous utiliserons la démonstration par récurrence pour démontrer $N(k) = 3k^2 - 2k - 1$ le plus grande nombre de cases de la grille de ce jeu qui sont dépendantes de la case :

Supposons que $N(k) = 3k^2 - 2k - 1$ est vrai pour SuDoku^k

si $k = 2$, alors pour SuDoku², chaque case dépend avec 7 cases

$$\text{D'un autre côté : } N(2) = 3 \times 2^2 - 2 \times 2 - 1 = 7$$

Donc $N(k)$ est vrai pour $k = 2$

Maintenant nous devons prouver $N(k + 1) = 3(k + 1)^2 - 2(k + 1) - 1$:

Nous avons déjà que chaque case dans une grille de SuDoku^{k+1} dépend de $(k + 1)^2 - 1$ cases de même bloc et $((k + 1)^2 - (k + 1))$ cases de même colonne et $((k + 1)^2 - (k + 1))$ cases de même ligne.

Après on somme de tous les résultats, Alors nous trouvons :

$$N(k) = 2((k + 1)^2 - (k + 1)) + (k + 1)^2 - 1$$

$$N(k) = 3(k + 1)^2 - 2(k + 1) - 1$$

1.3.2 Dénombrement les possibilités des solutions

Une grille de SuDoku³ correctement conçu a une et une seule solution : la grille finale est unique, mais la résolution à partir de la grille partielle peut toutefois prendre des chemins différents.

La question est combien de grilles de SuDoku³ y a-t-il ?

la réponse est un très grand nombre, difficile à trouver en énumérant toutes les possibilités

En 2005, **Bertram Felgenhauer** et **Frazer Jarvis** [2] ont prouvé que ce nombre de grilles était de :

$$6\ 670\ 903\ 752\ 021\ 072\ 936\ 960 \approx 6,67 \times 10^{21}$$

Ce nombre a été trouvé par brute force, la méthode est décrite dans [2].

Depuis ce premier calcul, plusieurs autres ont été créés pour compter le nombre de solutions possibles pour les variantes rectangulaires du SuDoku $n^2 \times m^2$, telle que $n \times m$ la taille de bloc de SuDoku rectangulaire. Ceux-ci sont donnés dans le tableau suivant [3] :

n	m	nombre
2	2	288
2	3	$28200960 \approx 2.8 \times 10^7$
2	4	$29136487207403520 \approx 2.9 \times 10^{16}$
2	5	$1903816047972624930994913280000 \approx 1.9 \times 10^{30}$
3	3	$6670903752021072936960 \approx 6,67 \times 10^{21}$
3	4	$81171437193104932746936103027318645818654720000 \approx 8.1 \times 10^{46}$
3	5	3.5086×10^{84}
4	4	5.9584×10^{98}
4	5	3.1764×10^{175}
5	5	4.3648×10^{308}

Grilles de solutions possibles pour les SuDokus rectangulaires

(Voir les variantes de SuDoku dans l'annexe)

1.3.3 Equivalences de grille SuDoku³

Les chiffres spécifiques utilisés pour remplir les cases différencient les grilles.

Il existe des éléments qui permettent de définir des grilles dites équivalentes, Cela soulève une question : combien y a-t-il de grille de SuDoku³ radicalement différents, notamment si on tient compte de certaines symétrie ou propriétés ?

Ces particularités se retrouvent dans la liste suivante :

1. Permutations des 9 nombres
2. Echange des lignes avec les colonnes (transposition).

3. Permutation des lignes dans une bande.
4. Permutation des colonnes dans une pilier.
5. Permutation des bandes.
6. Permutation des pilier .
7. Toute rotation ou réflexion de la grille.

c'ette opérations transforment toujours une grille valide en une autre grille valide.

Ainsi on peut déduire une définition précise de la notion de deux grilles radicalement différentes .

on dira que deux grilles SuDoku³ son radicalement différentes s'il n'existe pas aucune combinaison d'éléments de la liste d'écrite précédamment, permettant le passage de l'une à l'autre.

En tenant compte de ces symétries, **Jarvis et Russell** ont montré qu'il y avait 5 472 730 538 grilles radicalement différentes [4].

1.3.4 Existence de la solution

Une question importante se pose : est-ce que la solution existe toujours ?

La solution dépend directement du nombre de givens et de leur position, par ailleurs on peut trouver des grilles sans solution ou avec plusieurs solution. La difficulté de la solution dépend du nombre minimum de chiffres pré-remplis et aussi du bon choix de la suite des chiffres à ajouter à chaque étape.

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

Fig.1.8 :une grille 9×9 n'a pas de solutions.

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

Fig.1.9 : une grille 9×9 à une solution.

(Les solutions sont dans l'annexe)

1.3.5 Unicité de la solution

Le nombre de givens minimum, pour obtenir une solution lors de la résolution de la grille, n'a pas été déterminé. Jusqu'à présent aucune preuve mathématique n'a été trouvée. Mais des japonais ont déterminé des grilles de 17 givens qui donnent une unique solution [2].

Le nombre maximal de givens tel que la grille admet encore plus d'une solution est de $9^2 - 2^2$ c'est à dire 77 givens [4].

La solution d'une grille n'est pas toujours unique. Il existe des grilles à deux solutions différentes, et aussi des grilles qui n'admettent pas de solutions.

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

Fig.1.10 : une grille 9×9 à deux solutions différentes.

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

Fig.1.11 : une grille 9×9 à quatre solutions différentes.

(Les solutions sont dans l'annexe)

Chapitre 2

PROBLEMES DE SATISFACTION DE CONTRAINTE ET DE COLORATION DANS UN GRAPHE

2.1 Le Problème de Satisfaction de Contraintes (CSP)

2.1.1 Programmation linéaire

Un programme linéaire est un programme mathématique qui consiste à optimiser (maximiser ou minimiser) une fonction linéaire de plusieurs variables qui sont reliées par des équations linéaires et/ou des inéquations linéaires appelées contraintes.

Formellement, un programme linéaire consiste, étant donné un vecteur $C \in \mathbb{R}^n$, une matrice $A = A_{m \times n} = (a_{ij})_{\substack{1 \leq i \leq m \\ 1 \leq j \leq n}}$, $m, n \in \mathbb{N}^*$, à valeurs réelles, et un vecteur $b \in \mathbb{R}^m$, à trouver un vecteur $x \in \mathbb{R}^n$, solution du programme

**CHAPITRE 2. PROBLEMES DE SATISFACTION DE CONTRAINTES
ET DE COLORATION DANS UN GRAPHE**

mathématique suivant :

$$\left\{ \begin{array}{l} \text{Sous forme canonique} \\ z = \text{Maximiser (minimiser)} C^t x, \\ \text{Sous les contraintes} \\ Ax \leq b \text{ ou bien } Ax \geq b \end{array} \right. \quad \left\{ \begin{array}{l} \text{Sous forme standard} \\ z = \text{Maximiser (minimiser)} C^t x, \\ \text{Sous les contraintes} \\ Ax = b \end{array} \right.$$

Un programme linéaire en nombres entiers (PLNE) est dite écrite comme suit :

$$\left\{ \begin{array}{l} \text{Sous forme canonique} \\ z = \text{Maximiser (minimiser)} C^t x, \\ \text{Sous les contraintes} \\ Ax \leq b \text{ ou bien } Ax \geq b \\ x \in \mathbb{Z}^n \end{array} \right. \quad \left\{ \begin{array}{l} \text{Sous forme standard} \\ z = \text{Maximiser (minimiser)} C^t x, \\ \text{Sous les contraintes} \\ Ax = b \\ x \in \mathbb{Z}^n \end{array} \right.$$

Un programme linéaire en nombres entiers (PLNE) : est un programme qui ne contient que des variables entières.

Un programme linéaire en variables binaires(PLVB) • «qui est un cas particulier de la PLNE » ne contient que des variables booléennes.

2.1.2 Problème de satisfaction des contraintes (CSP)

Définitions

Un CSP est défini comme étant un ensemble de contraintes impliquant un certain nombre de variables. L'objectif consiste simplement à trouver un ensemble de valeurs à affecter aux variables, de sorte que toutes les contraintes soient satisfaites. Dans le cas le plus général, les problèmes de satisfaction des contraintes ont un aspect fortement combinatoire qui leur confère une grande complexité.

On peut définir un (csp) comme étant un problème caractérisé par un triplet (R, D, K) où :

- $R = (r_1, r_2, \dots, r_n)$ ensemble des variables de décision
- $D = (d_1, d_2, \dots, d_n)$ ensemble de domaines , d_i étant le domaine de la variable r_i , En d'autres termes , $\forall i = 1, \dots, n. r_i \in d_i$
- $K = (k_1, k_2, \dots, k_m)$ ensemble des contraintes, chaque contrainte k_i est une relation entre certaines variables de R , restreignant les valeurs que peuvent prendre simultanément ces variables.

2.1. LE PROBLÈME DE SATISFACTION DE CONTRAINTES (CSP)

Exemple 2.1.1 on peut définir le CSP (R, D, K) suivant :

- $R = \{a, b, c, d\}$
- $D(a) = D(b) = D(c) = D(d) = \{0, 1\}$
- $K = \{a \neq b, c \neq d, a + c < b\}$

Ce CSP comporte 4 variable a, b, c et d , chacune pouvant prendre deux valeurs (0 ou 1). Ces variables doivent respecter les contraintes suivantes : a doit être différente de b , c doit être différente de d et la somme de a et c doit être inférieure à b .

Les variables du CSP

Les variables d'un CSP sont généralement les inconnues du problème dont on cherche à associer des valeurs, Dans ce qui suit, nous étudierons exclusivement les CSP à domaines finis, où chaque domaine de valeur est un ensemble isomorphe à une partie finie de l'ensemble des entiers naturels \mathbb{N} . Un domaine de valeurs peut se représenter comme un ensemble d'entier naturels représentables en machines.

Solution d'un CSP

Etant donné un CSP (R, D, K) . Sa résolution consiste à affecter des valeurs aux variables, de telle sorte que toutes les contraintes soient satisfaites. On introduit pour cela les notations et définitions suivantes :

Définition 2.1.1 On appelle **affectation** le fait d'instancier certaines variables par des valeurs (évidemment prises dans les domaines des variables), On notera $A = \{(r_1, d_1), (r_2, d_2), \dots, (r_n, d_n)\}$ l'affectation qui instancie la variable r_1 par la valeur d_1 , la variable r_2 par la valeur d_2 ..., et la variable r_n par la valeur d_n ,

Exemple 2.1.2 par exemple, sur le CSP précédent, $A = \{(b, 0), (c, 1)\}$ est l'affectation qui instancie b à 0 et c à 1 .

Définition 2.1.2 Une affectation est dite **total** si elle instancie toutes les variables du problème, elle est dite **partielle** si elle n'en instancie qu'une partie.

Exemple 2.1.3 Dans notre exemple, $A_1 = \{(a, 1), (b, 0), (c, 0), (d, 0)\}$ est une affectation total, $A_2 = \{(a, 0), (b, 0)\}$ est une affectation partielle.

Définition 2.1.3 Une affectation A **viole** une contrainte k_i si toutes les variables de k_i sont instanciées dans A , et que la relation définie par k_i n'est pas vérifiée pour au moins une valeur des variables de k_i définies dans A .

Exemple 2.1.4 Dans notre exemple, l'affectation partielle $A_2 = \{(a, 0), (b, 0)\}$ viole la contrainte $a \neq b$, en revanche, elle ne viole pas les deux autres contraintes dans la mesure où certaines de leurs variables ne sont pas instanciées dans A_2 .

Définition 2.1.4 Une affectation (totale ou partielle) est **consistante** si elle ne viole aucune contrainte, et **inconsistante** si elle viole une ou plusieurs contraintes.

Exemple 2.1.5 Dans notre exemple, l'affectation partielle $\{(c, 0), (d, 1)\}$ est consistante, tandis que l'affectation partielle $\{(a, 0), (b, 0)\}$ est inconsistante.

Définition 2.1.5 Une **solution** est une affectation totale consistante, c'est-à-dire une valuation de toutes les variables du problème qui ne viole aucune contrainte.

Exemple 2.1.6 Dans notre exemple, $A = \{(a, 0), (b, 1), (c, 0), (d, 1)\}$ est une affectation totale consistante : c'est une solution

2.2 Quelques Rappels sur théorie des graphes

Dans ce travail, nous utilisons uniquement des graphes simples et non orienté.

Définition 2.2.1 Un **Graphe** non orienté $G = (Y, E)$ est définie par deux ensembles finis $Y = \{y_1, y_2, \dots, y_p\}$ dont les éléments sont appelés **Sommets** ($|Y| = p$), et $E = \{e_1, e_2, \dots, e_q\}$ l'ensemble de couples non ordonné de sommets, appelés **Arêtes** ($|E| = q$) et $E \subset Y^2$.

Définition 2.2.2 Deux sommets y_1 et y_2 sont dit **adjacents** si sont reliés par une arête $e = (y_1, y_2)$.

Définition 2.2.3 Le **degré** d'un sommet est le nombre d'arêtes reliant ce sommet, avec les boucles comptées deux fois. et on noté $\text{deg}(y)$.

2.2. QUELQUES RAPPELES SUR THÉORIE DES GRAPHS

Définition 2.2.4 On appelle **ordre** d'un graphe le nombre de ses sommets, i.e $\text{card}(Y)$.

Définition 2.2.5 Une **boucle** est une arête reliant un sommet à lui-même.

Définition 2.2.6 Un graphe non-orienté est dit **simple** s'il est sans boucle et présente une arête entre tout paire de sommets différents

Définition 2.2.7 Un graphe est dit régulier si tous ses sommets ont le même degré, un graphe régulier dont les sommets sont de degré k est appelé un graphe k -régulier ou graphe régulier de degré k .

Définition 2.2.8 Un graphe complet est un graphe simple dont les sommets sont adjacents deux à deux, c'est-à-dire que tout couple de sommets distincts sont reliés par une arête.

Définition 2.2.9 (Problème de coloration de sommées) Dans un graphe simple, la coloration consiste à attribuer une couleur à chaque sommet de manière à ce que deux sommets adjacents distincte soient de couleur différente. On appelle alors le nombre chromatique le nombre minimal de couleur nécessaire à la coloration des sommets. Ce nombre chromatique est désigné par $\chi(G)$

Chapitre 3

CHEMINEMENT DU JEU SuDoku³ AU PROBLEME DE SATISFACTION DE CONTRAINTES

3.1 Présentation du Problème de Satisfaction de Contraintes

Le problème de satisfaction de contraintes illustrant clairement les mécanismes de modélisation et de résolution d'un CSP sont les grilles de SuDoku. Cette section est consacrée à une présentation de l'approche CSP en nous appuyant sur la modélisation d'une grille de SuDoku

3.1.1 Variables dans les modèles SuDoku³

Dans ce modèle, une matrice cubique d'ordre 9 est définie avec un nombre total de $9 \times 9 \times 9$ éléments. Chaque variable ainsi définie est binaire et caractérisée par trois indices : i, j, h , les indices i et j définissent respectivement la ligne et la colonne d'une position générique dans une grille de SuDoku³, l'indice h représente un entier compris entre 1 et 9 pouvant être présent dans une position générique c'est à dire $v_{ij} = h$. Étant les variables binaires, ils ne peuvent prendre que les valeurs 0 ou 1 :

Soit la matrice cubique $X = (x_{ijh})_{i,j,h \in I_9}$ telle que :

3.1. PRÉSENTATION DU PROBLÈME DE SATISFACTION DE CONTRAINTES

$$x_{ijh} = \begin{cases} 1 & \text{si } v_{ij} = h \\ 0 & \text{sinon} \end{cases} \quad (3.1)$$

3.1.2 contraintes dans les modèles SuDoku

Il convient de s'assurer que les règles du jeu sont respectées.

contrainte sur ligne :

Chaque ligne doit contenir chaque valeur une et une seule fois,

Soit $X = (x_{ijh})_{i,j,h \in I_9}$ la matrice cubique .

Une ligne V_{i^*} fixée croise chaque colonne V_{*j} au point (v_{ij})

En utilisant (3.1) : si $v_{ij} = h, h \in I_9$

$$x_{ij_0h} = \begin{cases} 1 & \text{si } j_0 = j \\ 0 & \text{si } j \in I_9 \setminus \{j_0\} \end{cases}$$

ce qui donnerait :

$$\sum_{j=1}^9 x_{i,j,h} = 1$$

pour $h \in I_9$ donnée.

contrainte sur colonne :

Il en est de même pour chacune des colonnes,

Soit $X = (x_{ijh})_{i,j,h \in I_9}$ la matrice cubique .

Une colonne V_{*j} fixée croise chaque ligne V_{i^*} au point (v_{ij})

En utilisant (3.1) : si $v_{ij} = h, h \in I_9$

$$x_{i_0jh} = \begin{cases} 1 & \text{si } i_0 = i \\ 0 & \text{si } i \in I_9 \setminus \{i_0\} \end{cases}$$

ce qui donnerait :

$$\sum_{i=1}^9 x_{i,j,h} = 1$$

,pour $h \in I_9$ donnée.

contrainte sur bloc carrée :

Soit $X = (x_{ijh})_{i,j,h \in I_9}$ la matrice cubique .

$\forall (i_1, j_1) \in (\varphi(i_0, i), \varphi(j_0, j))$

CHAPITRE 3. CHEMINEMENT DU JEU SUDOKU³ AU PROBLEME
DE SATISFACTION DE CONTRAINTES

En utilisant (3.1) : si $v_{ij} = h$, $h \in I_9$

$$x_{i_1 j_1 h} = \begin{cases} 1 & \text{si } (i_1, j_1) \neq (i, j) \\ 0 & \text{si } i \in (\varphi(i_0, i), \varphi(j_0, j)) \setminus (i_1, j_1) \end{cases}$$

ce qui donnerait :

$$\sum_{i=1}^3 \sum_{j=1}^3 x_{\varphi(i_0, i), \varphi(j_0, j), h} = 1$$

contrainte Test d'arrêt : (Partie du jeu terminée)
chaque case ne doit contenir qu'une et une seule valeur

$$\sum_{h=1}^9 x_{i, j, h} = 1, \forall (i, j) \in I_9$$

i.e arrêt du jeu.

sinon

$$\text{si } \exists (i_1, j_1) \in I_9^2 / \sum_{h=1}^9 x_{i, j, h} = 0$$

i.e Le jeu est incomplet.

Remarque 3.1.1 Pour tout $i_0, j_0 \in I_3$, qui permettent de s'assurer que chaque valeur est présente une et une seule fois dans chacune des régions carrée, Après le remplacement de (i_0, j_0) dans l'équation de bloc carrée on trouve :

	$j_0 = 1$	$j_0 = 2$	$j_0 = 3$
$i_0 = 1$	$\sum_{i=1}^3 \sum_{j=1}^3 x_{i, j, h} = 1$	$\sum_{i=1}^3 \sum_{j=4}^6 x_{i, j, h} = 1$	$\sum_{i=1}^3 \sum_{j=7}^9 x_{i, j, h} = 1$
$i_0 = 2$	$\sum_{i=4}^6 \sum_{j=1}^3 x_{i, j, h} = 1$	$\sum_{i=4}^6 \sum_{j=4}^6 x_{i, j, h} = 1$	$\sum_{i=4}^6 \sum_{j=7}^9 x_{i, j, h} = 1$
$i_0 = 3$	$\sum_{i=7}^9 \sum_{j=1}^3 x_{i, j, h} = 1$	$\sum_{i=7}^9 \sum_{j=4}^6 x_{i, j, h} = 1$	$\sum_{i=7}^9 \sum_{j=7}^9 x_{i, j, h} = 1$

3.1. PRÉSENTATION DU PROBLÈME DE SATISFACTION DE CONTRAINTES

Résultat :

En tout, il y a :

$$9^3 = 729 \text{ variables}$$

$$4 \times 9^2 = 324 \text{ équations (contraintes).}$$

Le rang de la matrice variables-équations est 249, donc il n'y a que 249 équations linéairement indépendantes [5].

La fonction objectif

En ce point une "fonction objectif" être minimisé ou maximisé doit être défini. Mais il faut remarquer que dans notre cas, il suffit de déterminer une solution admissible, parce que chaque grille de Sudoku a une et une seule solution, alors la solution admissible est automatiquement la solution optimale.

ona :

$$\sum_{h=1}^9 x_{i,j,h} = 1 \implies \sum_{i=1}^9 \sum_{j=1}^9 \left(\sum_{h=1}^9 x_{i,j,h} \right) = \sum_{i=1}^9 \sum_{j=1}^9 1 = 9^2$$

Donc :

$$f(X) = \left[\frac{1}{9^2} \times \sum_{i=1}^9 \sum_{j=1}^9 \left(\sum_{h=1}^9 x_{i,j,h} \right) \right] = \left[\frac{1}{9^2} \times 9^2 \right] = 1$$

3.1.3 Énoncé du programme linéaire en nombres entiers

$$\left\{ \begin{array}{ll} \max & 1 \\ \sum_{h=1}^9 x_{i,j,h} = 1 & i, j \in I_9 \\ \sum_{j=1}^9 x_{i,j,h} = 1 & i, k \in I_9 \\ \sum_{i=1}^9 x_{i,j,h} = 1 & j, k \in I_9 \\ \sum_{i=1}^3 \sum_{j=1}^3 x_{\varphi(i_0,i),\varphi(j_0,j),h} = 1 & i_0, j_0 \in I_3 \end{array} \right.$$

3.2 Cheminement de SuDoku³ au CSP

Modéliser une grille de SuDoku³ en CSP revient à représenter les ensembles (R, D, K) de la manière suivante :

L'ensemble R :

Nous introduisons les 9^3 variables binaires : $X = x_{ijh}$ telle que :

$$x_{ijh} = \begin{cases} 1 & \text{si } v_{ij} = h \\ 0 & \text{sinon} \end{cases}$$

L'ensemble D :

$$D = \{D_X | x_{ijh} \in \{0, 1\}\}$$

L'ensemble K :

k_1 : **contrainte de ligne :**

Les valeurs dans chaque ligne sont différentes :

$$\sum_{j=1}^9 x_{i,j,h} = 1 \quad i, h \in I_9$$

3.3. EXEMPLE ($SUDOKU^2 \rightarrow CSP$)

k_2 : **contrainte de colonne** :

Les valeurs dans chaque colonne sont différents :

$$\sum_{i=1}^9 x_{i,j,h} = 1 \quad j, h \in I_9$$

k_3 : **contrainte de carrée** :

Les valeur dans chaque région carrée sont différents :

$$\sum_{i=1}^3 \sum_{j=1}^3 x_{\varphi(i_0,i),\varphi(j_0,j),h} = 1, \quad i_0, j_0 \in I_3, \forall h \in I_9$$

• k_4 **contrainte Test d'arrêt** :

$$\sum_{h=1}^9 x_{i,j,h} = 1 \quad i, j \in I_9$$

La satisfaction de l'ensemble des contraintes ci-dessus $k_i, 1 \leq i \leq 4$, aboutit alors à une **affectation totale et consistante**.

3.3 Exemple ($SuDoku^2 \rightarrow CSP$)

Soit une grille de SuDoku² de 11 givens suivant :

1	2	3	
4		2	1
	4	1	
2	1		3

Fig.2.1 SuDoku² de 11 givens

À partir de cette grille, nous remplissons la matrice cubique X étape par étape comme suit :

3.3.1 Etape 0 :

Niveau $h = 0$:

C'est la grille de SuDoku²

Niveau $h = 1$:

Contient les chiffres de valeurs 1 de la grille de niveau $h = 0$ mais remplacés par un 1 et le reste est tout égal à 0

Niveau $h = 2$:

Contient les chiffres de valeurs 2 de la grille de niveau $h = 0$ mais remplacés par un 1 et le reste est tout égal à 0

Niveau $h = 3$:

Contient les chiffres de valeurs 3 de la grille de niveau $h = 0$ mais remplacés par un 1 et le reste est tout égal à 0

Niveau $h = 4$:

Contient les chiffres de valeurs 4 de la grille de niveau $h = 0$ mais remplacés par un 1 et le reste est tout égal à 0

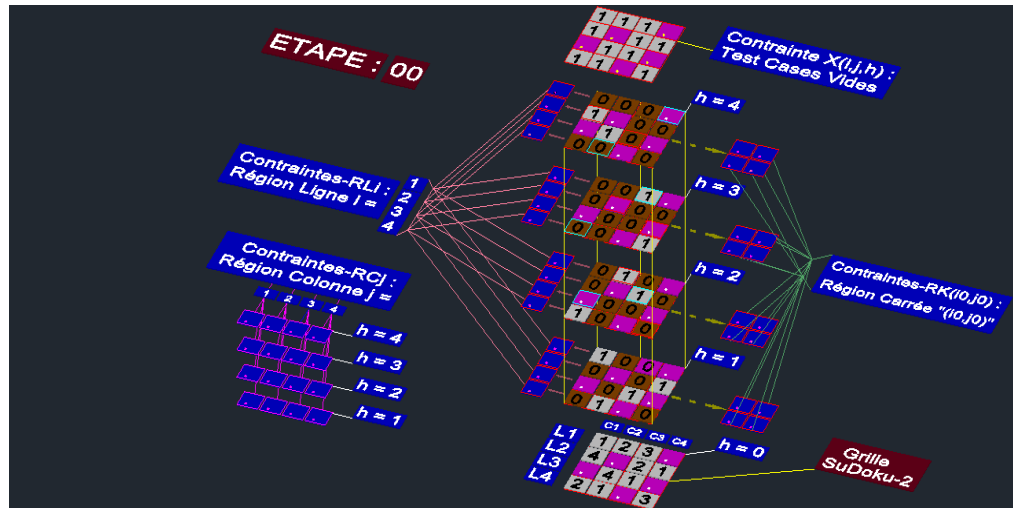


Fig.2.2 : Etape 00

3.3.2 Etape 1 :

Niveau $h = 0$:

On remplit la case qui dépend de L_1 et C_4 par la valeur 4

Niveau $h = 4$:

On remplit la case qui dépend de L_1 et C_4 par la valeur 1, et toutes les cases qui dépend d'elle par 0

3.3. EXEMPLE (SUDOKU² → CSP)

Les contraintes :

$$\sum_{j=1}^4 x_{1,j,h} = 1, \quad h \in I_4$$

$$\sum_{h=1}^4 x_{1,4,h} = 1$$

$$\sum_{i=1}^2 \sum_{j=1}^2 x_{\varphi(1,i),\varphi(2,j),h} = 1, \quad h \in I_4$$

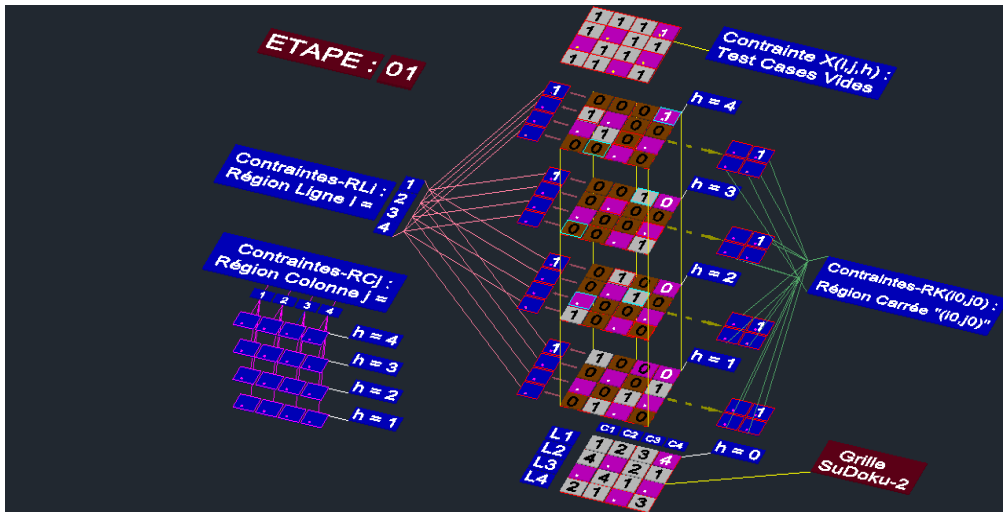


Fig.2.3 : Etape 01

3.3.3 Etape 2 :

Niveau $h = 0$:

On remplit la case qui dépend de L_2 et C_2 par la valeur 3

Niveau $h = 3$:

On remplit la case qui dépend de L_1 et C_4 par la valeur 1, et toutes les cases qui dépend d'elle par 0

Les contraintes :

$$\sum_{j=1}^4 x_{2,j,h} = 1, \quad h \in I_4$$

$$\sum_{i=1}^4 x_{i,2,h} = 1, \quad h \in I_4$$

$$\sum_{h=1}^4 x_{2,2,h} = 1$$

$$\sum_{i=1}^2 \sum_{j=1}^2 x_{\varphi(1,i),\varphi(1,j),h} = 1, \quad h \in I_4$$

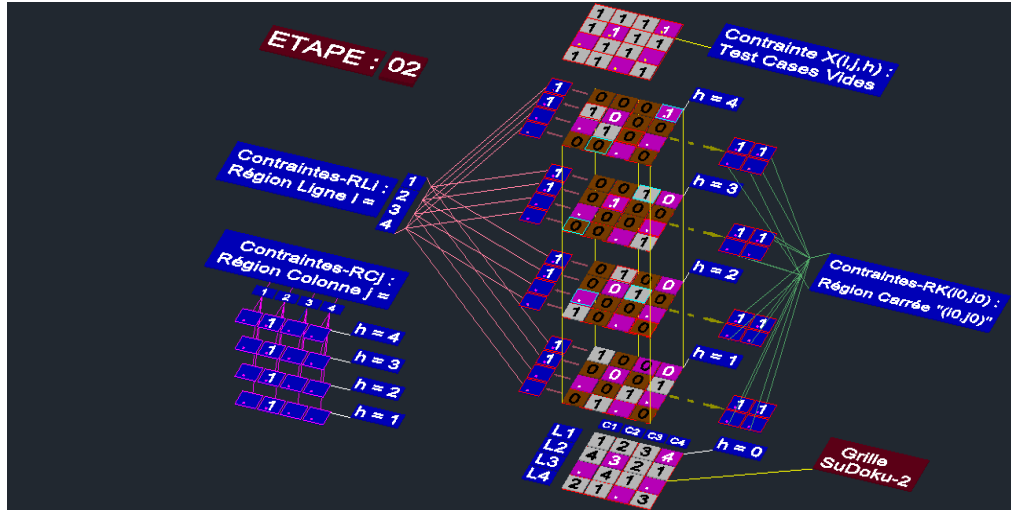


Fig.2.4 : Etape 02

3.3.4 Etape 3 :

Niveau $h = 0$:

On remplit la case qui dépend de L_3 et C_1 par la valeur 3

Niveau $h = 3$:

On remplit la case qui dépend de L_3 et C_1 par la valeur 1, et toutes les cases qui dépend d'elle par 0

Les contraintes :

$$\sum_{i=1}^4 x_{i,1,h} = 1, \quad h \in I_4$$

3.3. EXEMPLE (SUDOKU² → CSP)

$$\sum_{h=1}^4 x_{3,1,h} = 1$$

$$\sum_{i=1}^2 \sum_{j=1}^2 x_{\varphi(2,i),\varphi(1,j),h} = 1, \quad h \in I_4$$

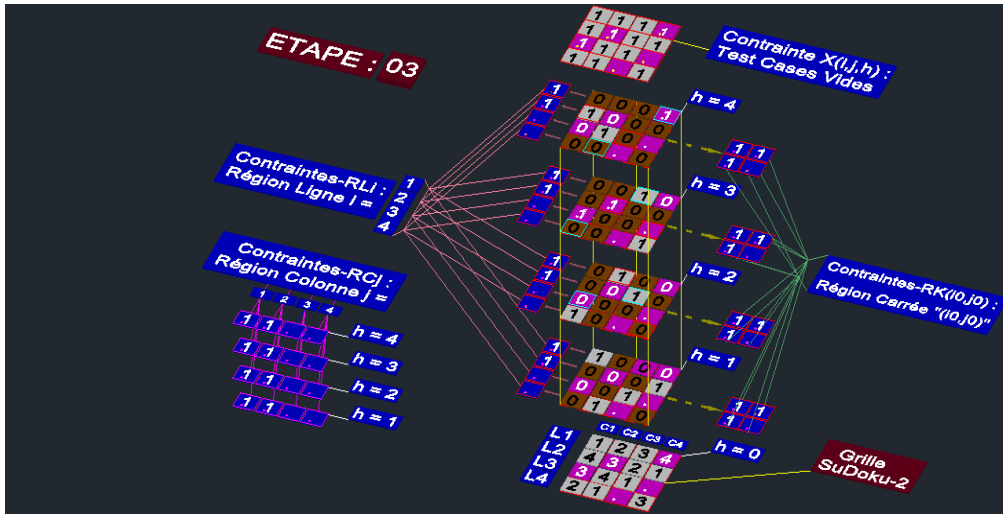


Fig.2.5 : Etape 03

3.3.5 Etape 4 :

Niveau $h = 0$:

On remplit la case qui dépend de L_4 et C_3 par la valeur 4

Niveau $h = 4$:

On remplit la case qui dépend de L_4 et C_3 par la valeur 1, et toutes les cases qui dépend d'elle par 0

Les contraintes :

$$\sum_{j=1}^4 x_{4,j,h} = 1, \quad h \in I_4$$

$$\sum_{i=1}^4 x_{i,3,h} = 1, \quad h \in I_4$$

$$\sum_{h=1}^4 x_{4,3,h} = 1$$

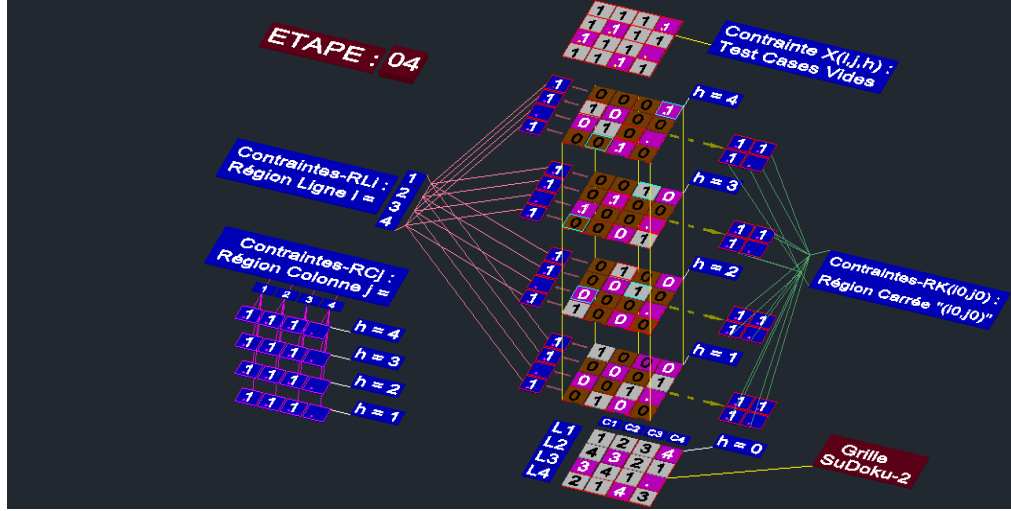


Fig.2.6 : Etape 04

3.3.6 Etape 5 :

Niveau $h = 0$:

On remplit la case qui dépend de L_3 et C_4 par la valeur 2

Niveau $h = 2$:

On remplit la case qui dépend de L_3 et C_4 par la valeur 1, et toutes les cases qui dépend d'elle par 0

Les contraintes :

$$\sum_{j=1}^4 x_{3,j,h} = 1, \quad h \in I_4$$

$$\sum_{i=1}^4 x_{i,4,h} = 1, \quad h \in I_4$$

$$\sum_{h=1}^4 x_{3,4,h} = 1$$

$$\sum_{i=1}^2 \sum_{j=1}^2 x_{\varphi(2,i),\varphi(2,j),h} = 1, \quad h \in I_4$$

3.3. EXEMPLE ($SUDOKU^2 \rightarrow CSP$)

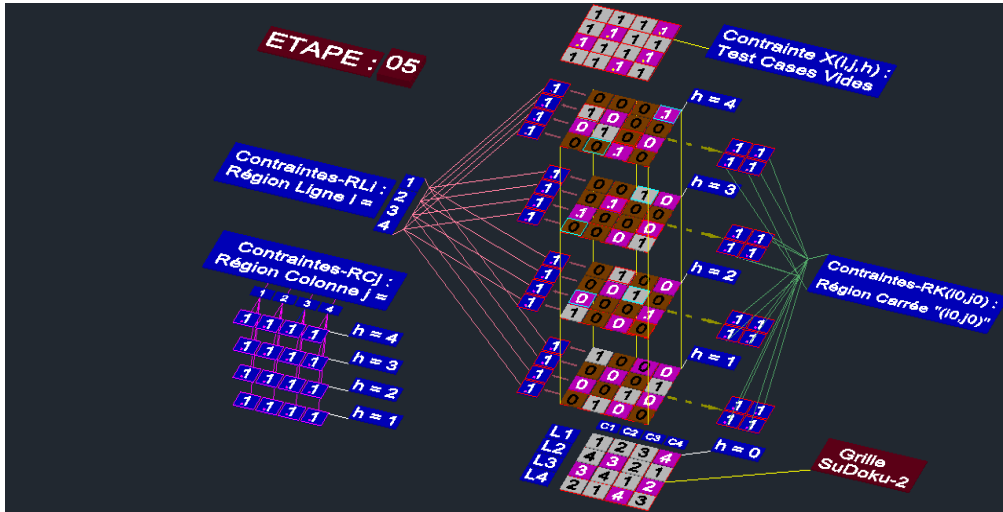


Fig.2.7 : Etape 05

Remarque 3.3.1 Ces graphiques ont été dessinés à l'aide du logiciel AutoCAD

Chapitre 4

CHEMINEMENT DU JEU SuDoku³ AU PROBLEME DE COLORATION DE SOMMETS

4.1 Introduction

Peut il y avoir des ressemblance entre le problème générale posé par un grille du jeu SuDoku³ et un problème posé dans un graphe .

Compte tenu des particularités de SuDoku³ déjà vues dans le chapitre 1 de ce document, on retient que :

Question 1

à quelles conditions un graphe noté $G = (Y, E)$ peut-il représenter la grille $C = (c_{ij})_{i,j \in I_9}$ du jeu SuDoku³

Question 2

Quelle élément de la grille SuDoku³ pourrait définir l'ensemble des sommets Y ?

Réponse 2

Les cases peuvent représenter les sommets d'un graphe $G = (Y, E)$

Dans ce cas la case c_{ij} représenterait le sommet y_{ij}

Question 3

Quelle élément de la grille SuDoku³ pourrait définir l'ensemble des arrêtes E ?

4.2. CHEMINEMENT DE SUDOKU³ À LA COLORATION DE

SOMMETS DANS UN GRAPHE : $G = (Y, E)$

Réponse 3

Les cases d'une même régions permettent de définir l'ensemble des arrêtes grace aux règles qui y sont définies.

Question 4

Quelle est la relation entre deux sommets $(y_{i_1, j_1}, y_{i_2, j_2}) \in Y$ pour que $(y_{i_1, j_1}, y_{i_2, j_2}) \in E$?

Réponse 4

le sommet $y_{i_1, j_1} \in Y$ est en relation avec $y_{i_2, j_2} \in Y$ si au moins l'une de ces conditions est vérifiée :

$$\forall (i_1, j_1) \neq (i_2, j_2) \in I_9^2$$

1. $\forall c_{i_1, j_1}, c_{i_2, j_2} \in RK_{i_0, j_0} \quad i_0, j_0 \in I_3$
2. $\forall c_{i_1, j_1}, c_{i_2, j_2} \in RL_i \quad \forall i \in I_9$
3. $\forall c_{i_1, j_1}, c_{i_2, j_2} \in RC_j \quad \forall j \in I_9$

Question 5

Peut-on avoir un couple $(y_{i_1, j_1}, y_{i_2, j_2})$ qui ne forme pas une arête ?

Réponse 5

si $(c_{i_1, j_1}, c_{i_2, j_2}) \in B_{i_0}$ et $c_{i_1, j_1} \in P_{j_0}, c_{i_2, j_2} \in P_{j'_0}$ tq $j_0 \neq j'_0$
 ou $(c_{i_1, j_1}, c_{i_2, j_2}) \in P_{j_0}$ et $c_{i_1, j_1} \in B_{i_0}, c_{i_2, j_2} \in B_{i'_0}$ tq $i_0 \neq i'_0$
 ou $c_{i_1, j_1} \in B_{i_0}, c_{i_2, j_2} \in B_{i'_0}$ et $c_{i_1, j_1} \in P_{j_0}, c_{i_2, j_2} \in P_{j'_0}$ tq $i_0 \neq i'_0, j_0 \neq j'_0$

4.2 Cheminement de SuDoku³ à la Coloration de Sommets dans un graphe : $G = (Y, E)$

4.2.1 Les étapes de coloration

Etape 1 :

Ecarter les colonnes puis les lignes de quelque millimètre de sorte à garder l'ordre des région ligne, région colonne ou région carrée.

L'ensemble des sommets est ainsi détermine :

$$Y = \bigcup_{i=1}^9 \left(\bigcup_{j=1}^9 y_{ij} \right)$$

Etape 2 :

Placer une arête entre tous les sommets d'une même région ligne, région colonne, région carrée. On définit ainsi l'ensemble des arêtes.

Faire ceci pour toutes les régions c'est à dire :

1. Les 9 régions lignes.
2. Les 9 régions colonnes.
3. Les 9 régions carrées.

Etape 3 :

Considérons 9 couleurs différentes, codifiées par les chiffres 1,2,3,4,5,6,7,8,9.

Etape 4 :

Placer les codes couleur sur les sommets non codés déjà de sorte que :

Deux sommets adjacents n'aient pas le même code couleur.

4.3 Exemple (SuDoku² \longrightarrow Coloration dans $G = (Y, E)$)

Soit la grille C de SuDoku² de 7 givens suivant :

2		3	
	1		
	3		2
4			3

Fig.3.1 : SuDoku² de 7 givens

4.3. EXEMPLE (SUDOKU² → COLORATION DANS $G = (Y, E)$)

On prende 4 couleur

Le chiffre 1 représente la couleur Marron.

Le chiffre 2 représente la couleur Jaune.

Le chiffre 3 représente la couleur Bleu.

Le chiffre 4 représente la couleur Rose.

Soit le graphe $G = (Y, E)$ corespondant de la grille C

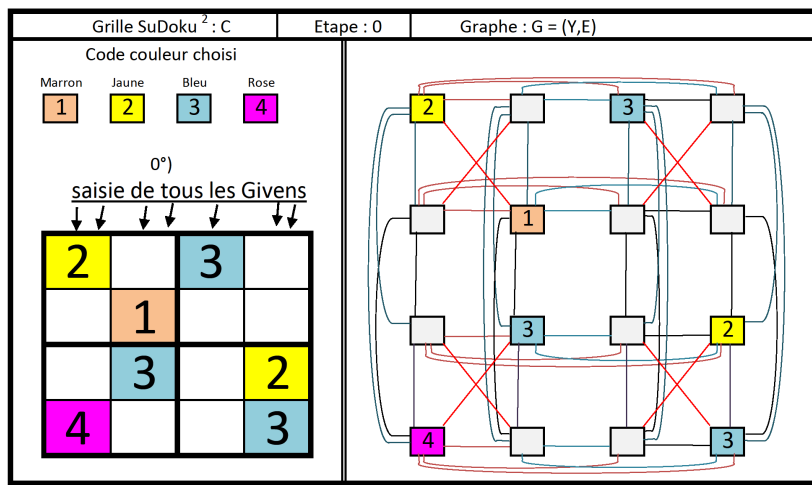


fig 3.2 : étape 0

On remplit la case c_{14} par la valeur 1 et on colorier le sommet que possed cette case par la couleur Marron

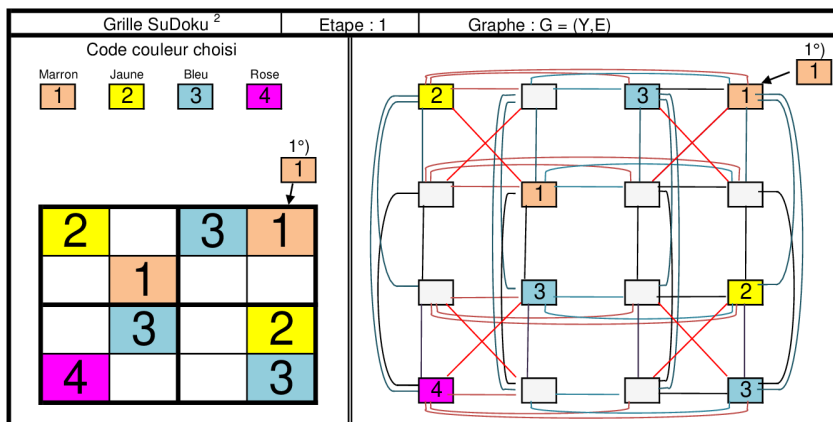


fig 3.3 : étape 1

CHAPITRE 4. CHEMINEMENT DU JEU SUDOKU³ AU PROBLEME
DE COLORATION DE SOMMETS

On remplit la case c_{12} par la valeur 4 et on colorier le sommet y_{12} par la couleur Rose.

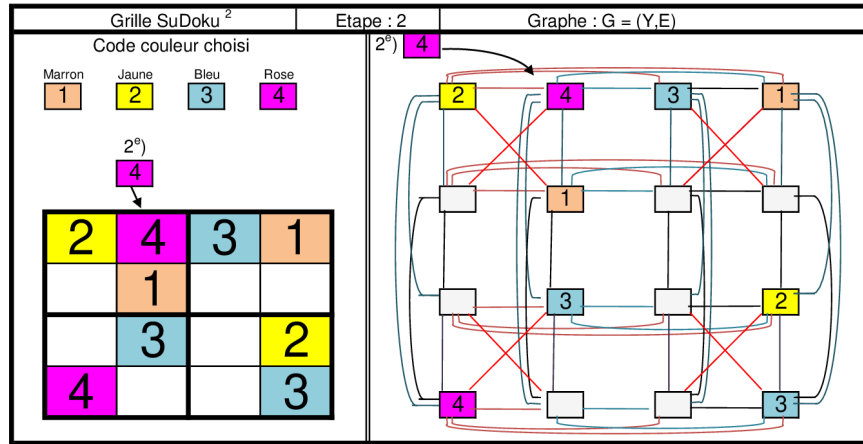


fig 3.4 : étape 2

On remplit la case c_{21} par la valeur 3 et on colorier le sommet y_{21} par la couleur Bleu.

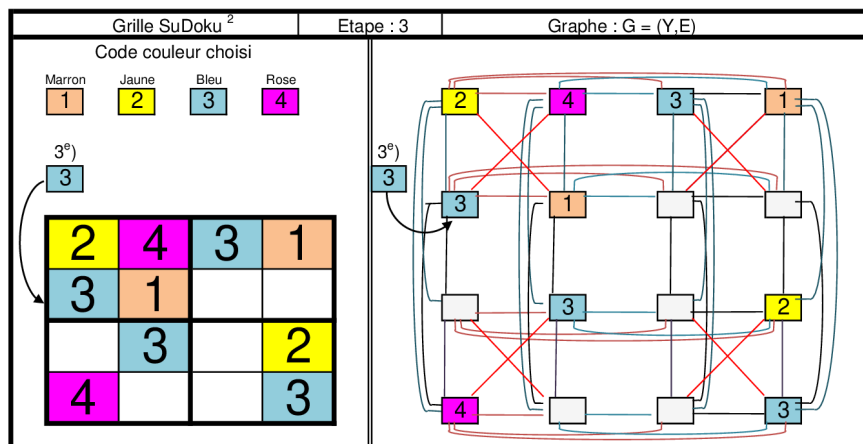


fig 3.5 : étape 3

On remplit la case c_{31} par la valeur 1 et on colorier le sommet y_{31} par la

4.3. EXEMPLE (SUDOKU² → COLORATION DANS $G = (Y, E)$)

couleur Marron.

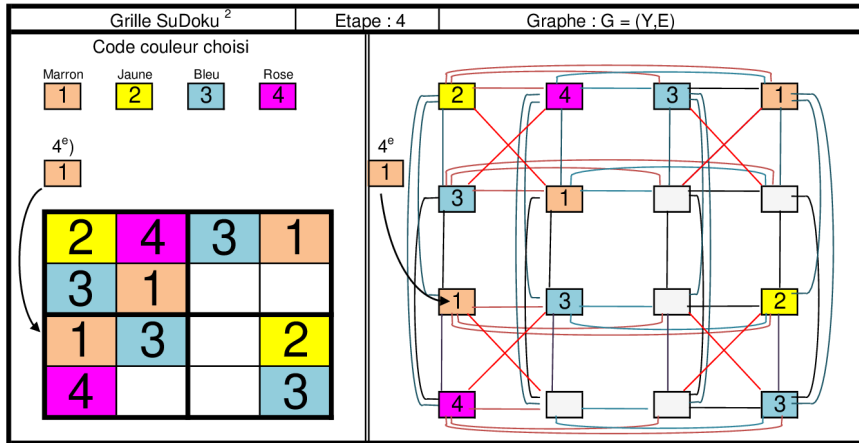


fig 3.6 : étape 4

On remplit la case c_{42} par la valeur 2 et on colorier le sommet y_{42} par la couleur jaune.

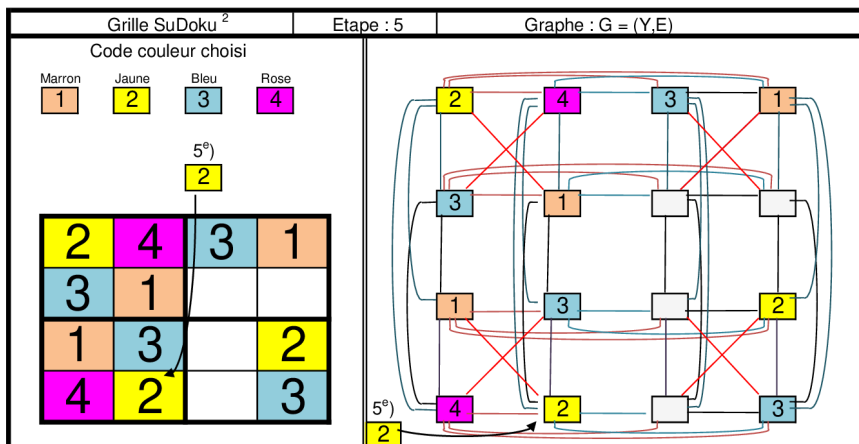


fig 3.7 : étape 5

On remplit la case C_{24} par la valeur 4 et on colorier le sommet y_{24} par la

CHAPITRE 4. CHEMINEMENT DU JEU SUDOKU³ AU PROBLEME DE COLORATION DE SOMMETS

couleur Rose.

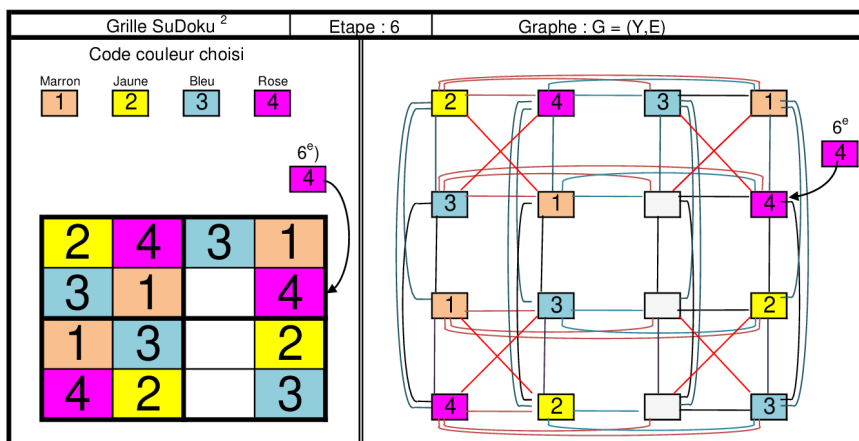


fig 3.8 : étape 6

On remplit la case c_{23} par la valeur 4 et on colorier le sommet y_{23} par la couleur jaune.

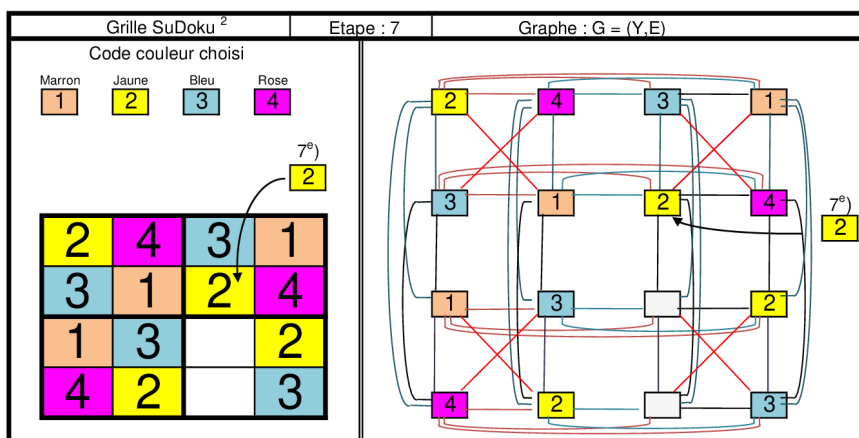


fig 3.9 : étape 7

On remplit la case C_{33} par la valeur 4 et on colorier le sommet y_{33} par la

4.3. EXEMPLE (SUDOKU² → COLORATION DANS $G = (Y, E)$)

couleur Rose.

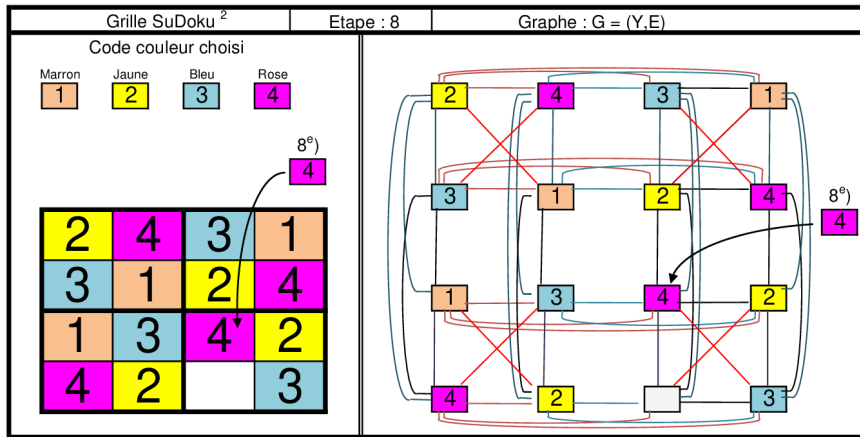


fig 3.10 : étape 8

On remplit la case C_{43} par la valeur 1 et on colorier le sommet y_{43} par la couleur Marron.

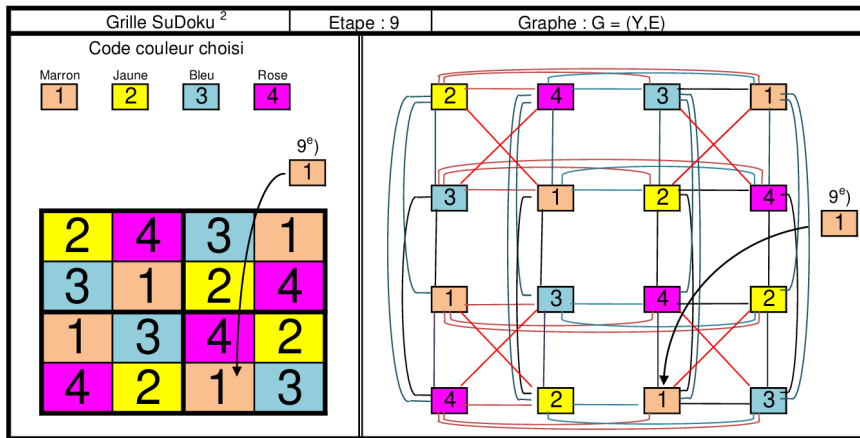


fig 3.11 : étape 9

Ici nous avons terminé le remplissage de la grille C et la coloration du graphe .

CHAPITRE 4. CHEMINEMENT DU JEU SUDOKU³ AU PROBLEME DE COLORATION DE SOMMETS

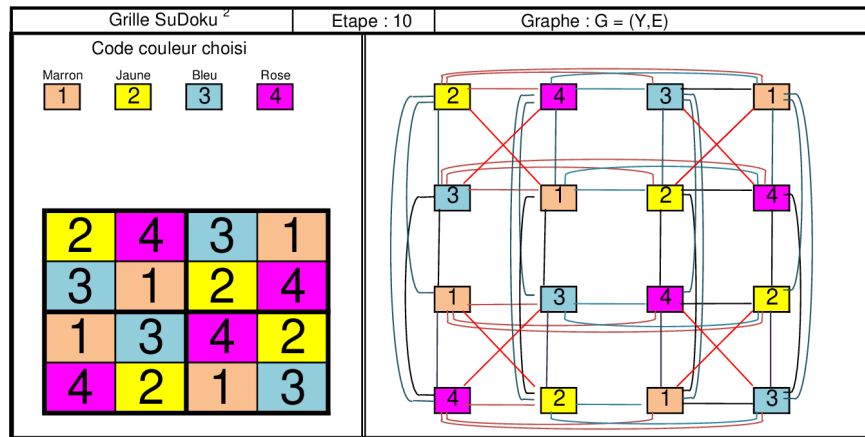


fig 3.12

Enfin voici le graphe $G = (Y, E)$ final qui correspondant à la grille C .

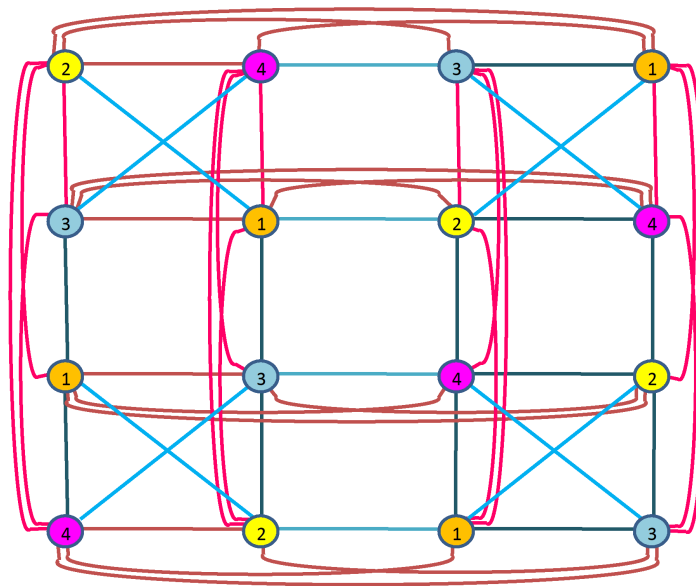


fig 3.12 : le graphe $G=(Y;E)$

Conclusion Générale

Résultats :

La problématique posée par le jeu SuDoku³ est équivalente, au moins par la complexité, à un problème de satisfaction de contraintes ou un problème de coloration de sommets dans un graphe.

Les problèmes de satisfactions de contraintes et de coloration de sommets dans un graphe étant classés de complexité difficile, on peut déduire que le jeu SuDoku³ est également un problème difficile.

Les deux modélisations proposées ont permis de déduire les résultats ci-dessus.

Après avoir analysé chaque méthode, nous concluons quelques points suivants :

Le jeu SuDoku^k de dimension $k^2 \times k^2$ modélisé en un CSP nous donne un programme linéaire de k^6 variables binaires et $4 \times k^4$ équations linéaires, notre travail sur (SuDoku³) c'est un programme linéaire de 3^6 variables et 4×3^4 équations linéaires.

De la même manière le jeu SuDoku^k de dimension $k^2 \times k^2$ modélisé en un problème de coloration de sommets dans un graphe nous donne un graphe $N(k)$ -régulier $G = (Y, E)$, telle que $(|Y| = k^4)$, $(|E| = \frac{N(k) \times k^4}{2})$, avec $N(k) = 3k^2 - 2k - 1$.

Chaque ligne, colonne et bloc représentent un sous-graphe complet de k^2 sommets donc :

$G = (Y, E)$ est un graphe admettant $3k^2$ sous graphes complets. Chaque sommet appartient à 3 sous-graphes.

notre travail sur (SuDoku³) c'est un graphe 20-régulier possédant 81 sommets et 810 arêtes. admettant $9 \times 3 = 27$ sous graphes complets.

Perspectives :

Après ce travail nous pouvons proposer qu'elle que question :

peut on donner un cheminement inverse. C'est à dire d'un problème de

*CHAPITRE 4. CHEMINEMENT DU JEU SUDOKU³ AU PROBLEME
DE COLORATION DE SOMMETS*

coloration de sommets vers le jeu SuDoku³

Remarque : une condition nécessaire pour telle travaille conservera le dimensionnement des deux modèles.

Annexe

(1) Nikoli :

est un éditeur japonais spécialisé dans les jeux et les casse-têtes. Son magazine le plus connu est le Puzzle Communication Nikoli (ou Nikoli). Nikoli devint populaire grâce au Sudoku. Les jeux publiés par Nikoli sont internationaux, car indépendants des langues et des cultures. Il faut dire que les casse-têtes japonais sont purement logiques et souvent numériques. Si l'alphabet diffère d'une culture à l'autre, le système numérique est international.

(2) Howard Garns :

(Mars 2,1905- 6 Octobre1989) était un architecte américain qui a gagné, seulement après sa mort, la renommée en tant que créateur de Number Place (« l'endroit des nombres »), un puzzle de nombres qui est devenu aujourd'hui un phénomène mondial sous le nom de SuDoku.

(3) Leonhard Euler :

né le 15 avril 1707 à Bâle et mort le 18 septembre 1783 à Saint Pétersbourg, est un mathématicien et physicien suisse, qui passa la plus grande partie de sa vie en Russie et en Allemagne. Il était notamment membre de l'Académie royale des sciences de Prusse à Berlin.

(4) Le carré magique :

En mathématiques, un carré magique d'ordre n est composé de n^2 entiers strictement positifs, écrits sous la forme d'un tableau carré. Ces nombres sont disposés de sorte que leurs sommes sur chaque rangée, sur chaque colonne et sur chaque diagonale principale soient égales. On nomme alors constante magique (et parfois densité) la valeur de ces sommes.

Un carré magique normal est un cas particulier de carré magique, constitué de tous les nombres entiers de 1 à n^2 , où n est l'ordre du carré.

(5) Le carré latin :

les cases d'un carré latin $n \times n$ sont remplies par des mêmes éléments (lettres, nombres, ... gures géométriques) distinctes en ligne et en colonne.

*CHAPITRE 4. CHEMINEMENT DU JEU SUDOKU³ AU PROBLEME
DE COLORATION DE SOMMETS*

On utilise donc n lettres (remarquer ici les permutations circulaires sur les lignes et les colonnes). Chaque ligne (ou colonne) s'obtient par permutation des n éléments. dans chaque ligne, chaque colonne, on ne trouve qu'une fois et une seule les n éléments.

La plupart du temps, les n éléments utilisés sont les entiers compris entre 1 et n , même si cela n'a aucune importance.

· les exemples des variantes :

2	1		
	3	2	
			4
1			

sudoku 4×4

	1			
			3	
		4		
1			5	
3				4

sudoku 5×5

	5				1
		4	6		
4				5	
1					4
	4	3			
6		2	4		

sudoku 6×6

		4				6
3		5				
		6	4			
			2	1		
	1					2
4				6	1	
7	6		3	4		

sudoku 7×7

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

sudoku 8×8

4.3. EXEMPLE (SUDOKU² → COLORATION DANS $G = (Y, E)$)

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

sudoku 9×9

9		8		1	10	14		2	13			4	11		
10		13			8	6	5	15	7			12	9		
4	7				3	16		8	14				2		
11	6		7	5		15		9	12			14	10	1	
	3		15	14			11		8	2				7	
	6	10		15	16		2	7	13			5		3	
				10	3		13	16				12	9	1	15
		1	14	6	8		5							16	4
		12		5	6	15		8	7	16			11	3	
		5			7		4	6	12		3			9	
14				3				15	2		9			5	12
	7		9		12			13	5		11		16	2	
7	11	8	12			9	3	1	4			14			5
	16					5		12	8	13	2	15			
6				13	10	1	11				5				8
	14	13	1	8		6			16	11	15	3		7	

sudoku 16×16

CHAPITRE 4. CHEMINEMENT DU JEU SUDOKU³ AU PROBLEME
DE COLORATION DE SOMMETS

1		4	25	19		10	21	8	14	6	12	9							5			
5		19	23	24	22	12		16	6	20	18	25	14	13	10	11			1	15		
		7	21	8	18		21	5	20	11	10	1	4	8	24	23	15	18	16	22	19	
						11		5	24			17	22	1	9	6	25					
		13	15	22	14	18	16		4			19				24	20	21	17			
12		11	6				15			21	25	19		4	22	14					20	
8			21	16			2	3			17	23	18	22							24	6
4		14	18	7	9	22	21	19		2	5				6	16	15				11	12
22		24	23		11	7			4	14	2	12		8	5	19					25	9
20				5			17	9	12	18	1		7	24							13	4
13			5		2	23	14	4	18	22	17		20	1	9	21	12				8	11
14	23		24							20	25	3	4	13	11	21	9	5			18	22
7			11	17	20	24			3	4	1	12		6	14	5	25	13				
		16	9	17	11	7	10	25		13	6		18			19	4					20
6	15	19	4	13		5	18	11		9	8	22	16	25	10	7						
			2		10	19	3	1	22	9	4	11	15	20			8	23				25
		24	8	13	1		4	20	17	14		18	16	22	5	11					10	
23	10					18	6	16		17	1		13			3	19	12				
25	5	14	11		17	8	24	13	19	23	15	9		12	20		22				7	
		17	4	22	15	23	11	12	25			18	8	7			14				13	
19	6	23	22	8		1	25	4	14	2	3	7	13	10	11	16						
		4		17	3	24	8	20	23	11	10	25	22			12	13	2	18	6		
			7	16		6	17	2	21	18		19		8							4	
18	9	25	1	2	11		13	22	4	21	5		23	7			15	3				8
	21	10		12	20	16	19					15	14	4	2	18	23	25	11	7		

sudoku 25×25

les grilles solution :

La grille définie dans la figure 1.8 n'a pas de solutions car la case c_{83} doit contenir 3 et 9 comme solutions, ce qui est impossible.

4.3. EXEMPLE (SUDOKU² \rightarrow COLORATION DANS $G = (Y, E)$)

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

solution de la grille définie dans la fig1.9

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

1^{ère} solution de la grille définie dans la fig1.10

CHAPITRE 4. CHEMINEMENT DU JEU SUDOKU³ AU PROBLEME
DE COLORATION DE SOMMETS

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

2^{ème} solution de la grille définie dans la fig1.10

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

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

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

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

Les quatre solutions de la grille présentée dans la fig 1.11

Bibliographie

- [1] <http://www.grilles-sudoku.com/le-sudoku.html>, Presentation du sudoku : l'origine et l'histoire du sudoku.
- [2] **Felgenhauer, Bertram, and Frazer Jarvis**, Enumerating possible Sudoku grids." Preprint available at <http://www.afjarvis.staff.shef.ac.uk/sudoku/sudoku.pdf> (2005).
- [3] **Jussien, Narendra**, A to Z of Sudoku. ISTE, 2007
- [4] **Hofer, Thomas**, Résolution du Sudoku. No. STUDENT. 2006.
- [5] **Sirdey, Renaud**, Sudokus et programmation linéairez." Quadrature 63 (2007) : 9-13.
- [6] **LALA, CHIRAAG RAMESH**, GRAPH THEORY OF SUDOKU.Diss. INDIAN INSTITUTE OF SCIENCE EDUCATION AND RESEARCH BHOPAL, 2013.