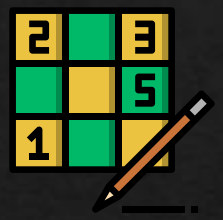
Solution de Norvig



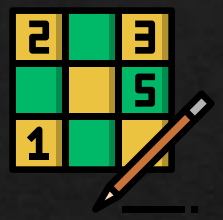
Sommaire

1. Peter Norvig, Un homme contre le Sudoku
2. Organisation
3. Résolution théorique
4. Notre implementation
5. Démonstration

Peter Norvig

- 63 ans
- Directeur de Recherche de Google (depuis 2006)
- Diplomé de l'université Brown (Maths appliquées) & Berkeley (informatique)
- Directeur de l'équipe de de recherche associé à Google Traduction

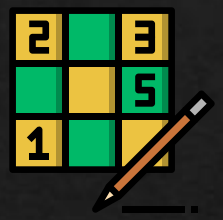
- Ben Laurie : "(Sudoku is) a denial of service attack on human intellect".
- "The goal was to solve every Sudolu Puzzle to be done with it".
- Solution en Python publié en 2011 "How to solve every Sudoku"



Organisation

Click to adOrganisation : A l'aide de l'interface ISudokuSolver, chaque groupe a pu dévelloper son solver sur une base commune, afin d'en faciliter le benchmark de manière centralisée
12 min
L'ensemble de notre code tient donc dans NorvigSolver, un sous-projet C# qui implémente l'interface mère.
Notre code n'utilise que les librairies système par défaut : using System; using System.Collections.Generic; using System.Text; using System.Linq; using System.Diagnostics; ainsi que Sudoku.Core; qui nous permet d'interagir avec les sudokus standardisés
Notre classe NorvigSolver n'étant instanciée qu'une seule fois par le benchmarker, nous avons placé les variables de son chargement (rows, cols, units, peers) en statique dans le constructeur, afin de ne les générer qu'une seule fois.
Cette optimisation a amélioré la vitesse de notre solveur de 30%.

A



Organisation

- Partie Implémentation :
  - Arnaud de Saint Méloire / Coumba Ndiaye / Pol De-Font-Reaulx

- Partie Documentation :
  - Stéphane Thurneyssen / Etienne Pinet

Résolution Théorique

- 1) Notation et Notions préliminaire

A1	A2	A3	A4	A5	A6	A7	A8	A9
B1	B2	B3	B4	B5	B6	B7	B8	B9
C1	C2	C3	C4	C5	C6	C7	C8	C9
D1	D2	D3	D4	D5	D6	D7	D8	D9
E1	E2	E3	E4	E5	E6	E7	E8	E9
F1	F2	F3	F4	F5	F6	F7	F8	F9
G1	G2	G3	G4	G5	G6	G7	G8	G9
H1	H2	H3	H4	H5	H6	H7	H8	H9
I1	I2	I3	I4	I5	I6	I7	I8	I8

- Un puzzle est résolu si les carrées dans chaque unité sont remplis avec une permutation des chiffres 1 à 9

Carré (81 existants)

Unité (27 existantes)

- Tout les carrés dans une même unités sont appelés « paires »
- Un carré a 3 unités et 20 paires

Résolution Théorique

- 2) Propagation de contraites

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

- Python

Les possibiltés sont vues pour chaques carrés comme suivant

values1 = {...,'E6': '4', 'E7':'14','E8':13456, ...},

value

- Si un carré n'a qu'une seule valeur possible, on élimine cette valeurs de ses carrés pairs

{...,'E6': '4', 'E7':'1','E8':1356, ...},

- Si une unite n'as qu'un seul carré possible pour une valeur, mettre assigné la valeur à ce carré

7	2	9	5	6	0	1	3	8

Résolution Théorique

- 2) Propagation de contraites

Avantages :

- Résoult rapidement les grilles faciles

Inconvénients :

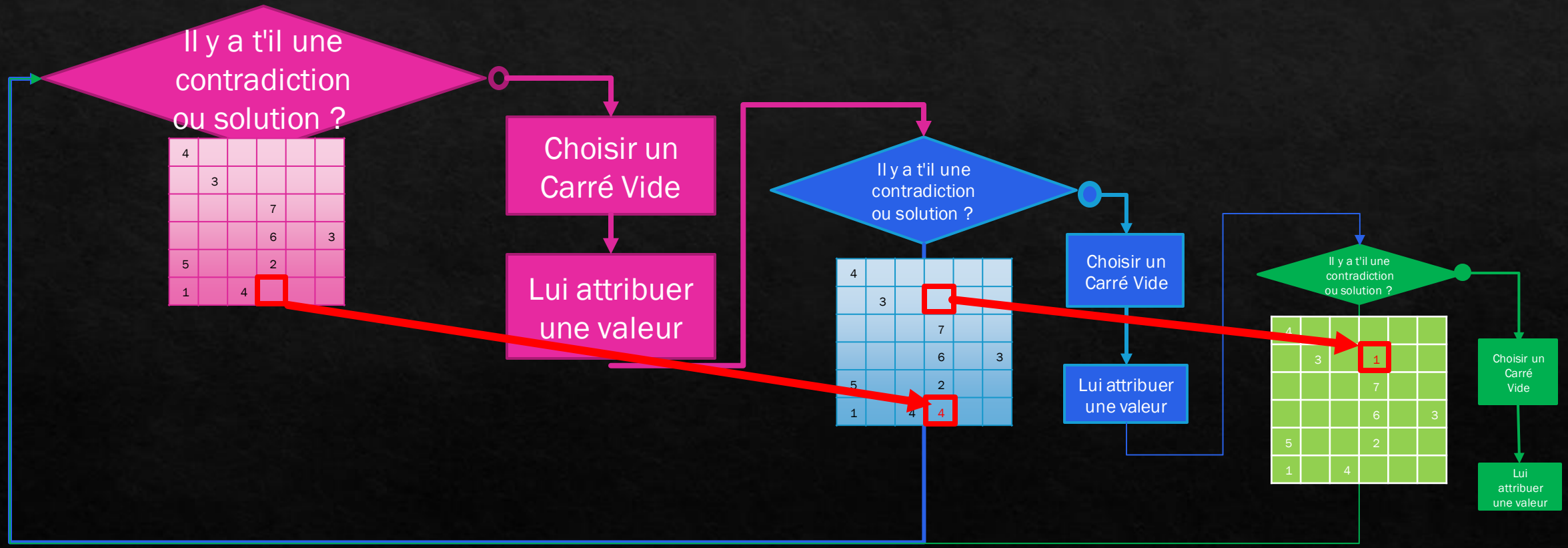
- N'est pas capable de résoudre certaines grilles si une situation de "Naked Twins" apparait.

Dans cette situation, il y aura 61 cases non résolus

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

Résolution Théorique

- 3) Fonction Search



Résolution Théorique

- 3) Fonction Search
  - Recursive

    On copie les valeurs possible entre chaque iteration

    On stock les “values” séparement

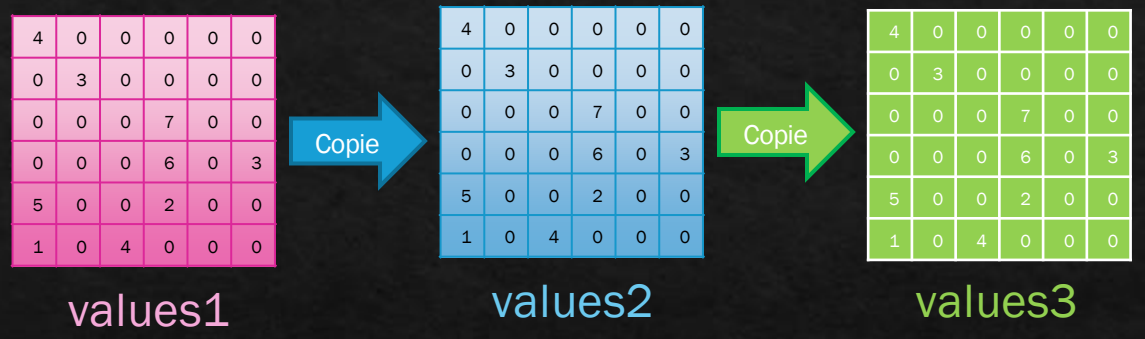
  - Nécessite 2 fonctions supplémentaire :
    - Variable Ordering

      “Quel Carré doit-être choisit en premier ?”

      On recherche le carré avec le - de “value” possible

    - Values Ordering :

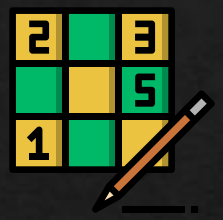
      1>2>3>4>5>6>7>8>9



values1 values2 values3

values :{...,’E6’: ‘1234', ‘E7':'14’,’E8’:13456, ...},

Chance d’erreur: {...,’E6’: ‘3/4’, ‘E7':’1/2’,’E8’:’5/6’, ...},



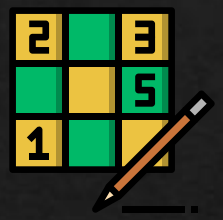
Résolution Théorique

- Complexité de l'Algorithme :
  - Plus la taille du tableau augmente, plus le temps de résolution augmente de manière exponentiellel
- Limite de la solution :
  - Utilisable uniquement pour le sudoku
  - Nécessité de transmettre au code les grilles sous formes particulière

Retour d'expérience : Workflow

- Notre IDE : **Visual Studio 2019**
  - Intégration de Git agréable
  - La gestion des sous-projets simplifie le travail entre groupes

- Notre gestionnaire de version : **GitHub**
  - Notre premier projet multi-équipes
  - Relecture des Pull-requests en ligne très utile
  - Gestion des permissions



“Merci de votre attention”

Des questions ?