

Utilizando Simulated Annealing para Resolver Sudoku

Aluno: Lyncoln Sousa de Oliveira



O que é o Sudoku?

- Origem Japonesa
- Quebra-cabeças lógico
- Somente 3 regras
- 8.388×10^{23} Possibilidades

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

Como resolver Sudoku computacionalmente?

- Simulated Annealing
 - Inspiração surge do processo físico do recozimento
 - Algoritmo de otimização
 - Aplicação utiliza métodos de Monte Carlo e Cadeias de Markov (MCMC)

O que é necessário para utilizar Simulated para otimização?

- Definir soluções como estados de uma cadeia de Markov
- Definir um estado inicial
- Definir uma função de custo
- Definir como um estado vai transicionar para um novo

Como fica para o Sudoku?

- Estado Inicial

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

- Função de Custo

- Elementos diferentes nas linhas
- Elementos diferentes nas colunas
- Máximo: 162

- Transição

- Sorteio de submatriz
- Sorteio de dois elementos
- Troca

Algoritmo

```
1. atual <- gera_estado_sudoku(tabuleiro)
2.
3. contador <- 0
4. maximo_iteracoes <- valor
5. while(contador < maximo_iteracoes){
6.   candidato <- transita_novo_estado(atual)
7.   funcao_custo_atual <- funcao_custo(atual)
8.   funcao_custo_candidato <- funcao_custo(candidato)
9.   delta_funcao_custo = funcao_custo_candidato - funcao_custo_atual
10.  if(delta_funcao_custo > 0){
11.    atual <- candidato
12.  } else { if(runif(1) < exp(delta_funcao_custo/temperatura)){
13.    atual <- candidato
14.  }
15. }
16. if(funcao_custo(atual) == 162) return(atual)
17. temperatura = temperatura*0.99999
18. contador = contador + 1
19. }
```

Resolvendo Sudoku

```
## [1] "Melhor métrica: 158 Metrica atual: 156 Iteração: 1000 Temperatura 0.495029842421621"  
## [1] "Melhor métrica: 160 Metrica atual: 160 Iteração: 2000 Temperatura 0.49010418868505"  
## [1] "Melhor métrica: 160 Metrica atual: 154 Iteração: 3000 Temperatura 0.485227546265887"  
## [1] "Melhor métrica: 160 Metrica atual: 154 Iteração: 4000 Temperatura 0.480399427490947"  
## [1] "Melhor métrica: 160 Metrica atual: 156 Iteração: 5000 Temperatura 0.475619349539502"  
## [1] "Melhor métrica: 160 Metrica atual: 156 Iteração: 6000 Temperatura 0.470886834394994"  
## [1] "Melhor métrica: 160 Metrica atual: 150 Iteração: 7000 Temperatura 0.46620140879723"  
## [1] "Melhor métrica: 160 Metrica atual: 148 Iteração: 8000 Temperatura 0.461562604195062"  
## [1] "Melhor métrica: 160 Metrica atual: 153 Iteração: 9000 Temperatura 0.456969956699524"  
## [1] "Melhor métrica: 160 Metrica atual: 154 Iteração: 10000 Temperatura 0.452423007037447"  
## [1] "Melhor métrica: 160 Metrica atual: 156 Iteração: 11000 Temperatura 0.447921300505529"  
## [1] "Melhor métrica: 160 Metrica atual: 158 Iteração: 12000 Temperatura 0.443464386924863"  
## [1] "Melhor métrica: 160 Metrica atual: 154 Iteração: 13000 Temperatura 0.43905182059592"  
## [1] "Melhor métrica: 160 Metrica atual: 154 Iteração: 14000 Temperatura 0.434683160253976"  
## [1] "Melhor métrica: 160 Metrica atual: 155 Iteração: 15000 Temperatura 0.430357969024989"  
## [1] "Melhor métrica: 160 Metrica atual: 157 Iteração: 16000 Temperatura 0.426075814381906"  
## [1] "Sudoku resolvido em 16609 passos!"
```

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

Avaliação do algoritmo

- Avaliação para Sudoku com $N=10, 15, \dots, 60$ Espaços em branco

N	Solucoes	Media Iteracoes	Desvio
10	50	44.50	34.14809
15	50	119.62	98.23605
20	50	94.58	63.82536
25	50	213.90	177.56565
30	50	321.38	243.91224
35	50	742.18	507.00182
40	50	1774.78	1724.45329
45	50	5037.28	5569.46390
50	48	12687.90	10795.98206
55	50	21539.78	17225.95508
60	50	20630.64	11442.17284

