

## Pengaplikasian Pewarnaan Graf dan Algoritma *Backtracking* pada Permainan Sudoku

Vanka Yuvadha Suciningtyas<sup>1\*</sup>, Zahrotul Janah<sup>2</sup>, Oryza Embun Putri Santosa<sup>3</sup>, Muhamad Gani Rohman<sup>4</sup>

<sup>1,2,3,4</sup>Universitas Islam Negeri Salatiga, Jalan Lingkar Selatan Salatiga KM 02 Pulutan Sidorejo Salatiga 50716, Indonesia E-mail:

<sup>1</sup>vankasuciningtyas@gmail.com, <sup>2</sup>zahrotul276@gmail.com, <sup>3</sup>oryzaembun@gmail.com, <sup>4</sup>muhamad.ganirohman@uinsalatiga.ac.id

\*Corresponding Author

### ABSTRACT

*Sudoku is a logic puzzle that can be mathematically modeled as a combinatorial problem in graph theory. This study aims to formulate the classic 9×9 sudoku game into a graph model and analyze its solution using a graph coloring approach and a backtracking algorithm. This study uses a theoretical study with a graph structural analysis approach and an algorithmic formulation. Each cell in the sudoku grid is represented as a node, while the neighbor relationships based on rows, columns, and 3×3 subgrids are represented as edges. The analysis results show that the sudoku graph is a regular graph of degree 20 and has a chromatic number  $\chi(G)=9$ . The sudoku problem is equivalent to a graph coloring using nine colors without any neighbor conflicts. In addition, the complexity analysis shows that the backtracking algorithm has exponential time complexity in the worst case, but the structure of the sudoku graph containing many maximum-sized cliques causes the pruning process to occur earlier. Thus, the integration of graph structural analysis and backtracking algorithms provides a systematic mathematical framework for explaining the characteristics of sudoku solutions.*

**Keywords:** Backtracking Algorithm, Graph, Graph Coloring, Sudoku

### ABSTRAK

Sudoku merupakan teka-teki logika yang secara matematis dapat dimodelkan sebagai permasalahan kombinatorial dalam teori graf. Penelitian ini bertujuan untuk memformulasikan permainan sudoku klasik berukuran 9×9 ke dalam model graf serta menganalisis penyelesaiannya melalui pendekatan pewarnaan graf dan algoritma backtracking. Penelitian ini menggunakan kajian teoretis dengan pendekatan analisis struktural graf dan formulasi algoritmik. Setiap sel pada grid sudoku direpresentasikan sebagai simpul, sedangkan hubungan ketetanggaan berdasarkan baris, kolom, dan subgrid 3×3 direpresentasikan sebagai sisi. Hasil analisis menunjukkan bahwa graf sudoku merupakan graf reguler berderajat 20 dan memiliki bilangan kromatik  $\chi(G)=9$ . Permasalahan sudoku ekuivalen dengan pewarnaan graf menggunakan sembilan warna tanpa konflik ketetanggaan. Selain itu, analisis kompleksitas menunjukkan bahwa algoritma backtracking memiliki kompleksitas waktu eksponensial dalam kasus terburuk, namun struktur graf sudoku yang memuat banyak klik berukuran maksimum menyebabkan proses pemangkasan (pruning) terjadi lebih awal. Dengan demikian, integrasi analisis struktural graf dan algoritma backtracking memberikan kerangka matematis yang sistematis dalam menjelaskan karakteristik penyelesaian sudoku.

**Kata kunci:** Algoritma Backtracking, Graf, Pewarnaan Graf, Sudoku

Dikirim: Juni 2025; Diterima: Juli 2025; Dipublikasikan: September 2025

Cara citasi: Suciningtyas, V. Y., Janah, Z., Santosa, O. E. P., & Rohman, M. G. (2025). Pengaplikasian Pewarnaan Graf dan Algoritma Backtracking pada Permainan Sudoku. *Teorema: Teori dan Riset Matematika*, 10(02), 203-212. DOI: <https://dx.doi.org/10.25157/teorema.v10i1.19774>.

This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license



## PENDAHULUAN

Sudoku merupakan teka-teki logika berbasis angka yang secara matematis dapat dimodelkan sebagai masalah kombinatorial. Sudoku mulai populer pada era 1970-an. Permainan ini mengharuskan pemain mengisi kotak-kotak kosong dengan angka 1 hingga 9 secara acak. Sudoku juga bermanfaat untuk mengembangkan kemampuan kognitif seseorang dalam proses pembelajaran matematika (Zaturrahmi, 2017).

Sudoku merupakan permainan digital yang tergolong dalam genre *puzzle*. Umumnya, sudoku dikenal sebagai teka-teki angka berbasis logika yang menantang pemain untuk mengisi kotak-kotak dalam *puzzle* dengan angka secara acak. Ukurannya bervariasi, mulai dari  $3 \times 3$ ,  $4 \times 4$ , hingga  $9 \times 9$ . Dalam permainan ini, dibutuhkan ketelitian, kesabaran, dan kemampuan logika untuk menyelesaikannya. Secara umum, bentuk sudoku terdiri dari matriks berukuran sembilan baris dan sembilan kolom yang dibagi menjadi blok-blok kecil berukuran  $3 \times 3$  (Rahman & Anubhakti, 2020).

Dalam perspektif teori graf, setiap sel pada grid sudoku dapat direpresentasikan sebagai simpul (vertex), sedangkan keterkaitan antar sel yang berada dalam baris, kolom, atau subgrid yang sama direpresentasikan sebagai sisi (edge). Representasi ini menghasilkan suatu graf tak berarah yang memiliki keteraturan struktur tinggi dan kendala pewarnaan yang ketat. Permasalahan penyelesaian sudoku kemudian ekuivalen dengan masalah pewarnaan graf dengan jumlah warna minimum tertentu yang memenuhi seluruh batasan ketetanggaan. Metode ini memungkinkan penggunaan berbagai teori graf secara sistematis dalam menyelesaikan Sudoku (Ayusi et al., 2025).

Studi sebelumnya menunjukkan bahwa pewarnaan graf adalah metode yang berpotensi efektif untuk menyelesaikan masalah sudoku. Sebagaimana dijelaskan oleh (Qomaruddin et al., 2022), tujuan pewarnaan graf adalah untuk memberikan warna pada titik-titik dalam graf, dengan syarat bahwa dua titik yang berdekatan tidak boleh memiliki warna yang sama. Menurut (Fransiskus Fran, 2019), konsep bilangan kromatik  $\chi(G)$ , yang merupakan jumlah warna minimal yang diperlukan menjadi dasar metode ini. Namun, dalam praktiknya, kinerja komputasi masih menjadi masalah, terutama untuk Sudoku dengan tingkat kesulitan tinggi.

Keterbatasan ini dapat diatasi dengan algoritma *backtracking*. Algoritma *backtracking* bekerja dengan mengevaluasi berbagai solusi potensial dan melakukan pruning terhadap rute yang tidak mengarah pada solusi yang benar, menurut (Bangun, 2024). Dalam hal penyelesaian Sudoku, kombinasi pewarnaan graf dan algoritma *backtracking* belum banyak dieksplorasi. Hal ini menciptakan gap penelitian yang bisa dikaji lebih dalam (Danuputri & Santosa, 2021).

Secara teoretis, graf sudoku memiliki struktur khusus yang berbeda dari graf umum, terutama dalam hal derajat simpul, keteraturan subgraf, dan keterkaitan lokal-global. Analisis mendalam terhadap sifat-sifat graf ini berpotensi memberikan landasan matematis yang lebih kuat dalam merumuskan strategi penyelesaian berbasis pewarnaan yang lebih efisien. Oleh karena itu, diperlukan kajian yang tidak hanya mengimplementasikan algoritma, tetapi juga menganalisis sifat struktural graf sudoku serta implikasinya terhadap proses pencarian solusi.

Berdasarkan uraian tersebut, penelitian ini bertujuan untuk:

1. Merepresentasikan permainan sudoku ke dalam model graf secara formal;
2. Menganalisis sifat-sifat struktural graf sudoku, termasuk derajat simpul dan keterkaitannya dengan bilangan kromatik;
3. Mengkaji penerapan pewarnaan graf dalam penyelesaian sudoku secara teoritis; dan
4. Mengintegrasikan pendekatan pewarnaan graf dengan algoritma *backtracking* dalam kerangka analisis matematis.

Kebaruan (novelty) penelitian ini terletak pada analisis terintegrasi antara struktur graf sudoku dan mekanisme pencarian solusi berbasis *backtracking* dalam satu kerangka matematis yang sistematis, sehingga diperoleh pemahaman yang lebih mendalam mengenai karakteristik penyelesaian sudoku dari sudut pandang teori graf.

## METODE PENELITIAN

Penelitian ini merupakan kajian teoretis dengan pendekatan analisis graf dan algoritma pencarian. Objek kajian berupa struktur matematis permainan sudoku klasik berukuran  $9 \times 9$  yang dimodelkan dalam bentuk graf tak berarah. Misalkan  $S$  adalah grid sudoku berukuran  $9 \times 9$ . Setiap sel pada grid direpresentasikan sebagai simpul pada suatu graf  $G = (V, E)$ , dengan  $V$  adalah himpunan simpul yang merepresentasikan seluruh sel sudoku dan  $E \subseteq V \times V$  adalah himpunan sisi yang menghubungkan

dua simpul jika dan hanya jika kedua sel tersebut berada pada baris, kolom, atau subgrid  $3 \times 3$  yang sama.

Dengan konstruksi tersebut, graf sudoku memiliki  $|V| = 81$  simpul. Derajat setiap simpul dianalisis berdasarkan jumlah keterhubungan yang timbul akibat kendala baris, kolom, dan subgrid. Selanjutnya, permasalahan penyelesaian sudoku diformulasikan sebagai masalah pewarnaan graf, yaitu menentukan fungsi pewarnaan  $f: V \rightarrow \{1, 2, \dots, 9\}$  sehingga untuk setiap pasangan simpul bertetangga  $u, v \in V$ , berlaku  $f(u) \neq f(v)$ .

Analisis dilakukan melalui dua tahap. Tahap pertama adalah kajian sifat struktural graf sudoku, meliputi derajat simpul, keteraturan lokal, dan implikasinya terhadap bilangan kromatik  $\chi(G)$ . Tahap kedua adalah formulasi algoritma backtracking sebagai prosedur konstruktif untuk menemukan fungsi pewarnaan yang memenuhi seluruh kendala ketetanggaan. Algoritma backtracking diformulasikan sebagai prosedur rekursif yang memilih simpul belum terwarnai, menetapkan warna yang memenuhi kendala lokal, dan melakukan penelusuran mundur apabila terjadi konflik pewarnaan, hal ini sesuai dengan penelitian Hanafi et al. (2021) yang menyatakan bahwa algoritma *backtracking* diterapkan untuk menyelesaikan pewarnaan graf yang sesuai dengan ketentuan sudoku. Kompleksitas waktu dianalisis berdasarkan ukuran ruang solusi dan jumlah kemungkinan pewarnaan parsial yang mungkin terbentuk selama proses pencarian.

Kajian ini tidak hanya memformulasikan prosedur algoritmik, tetapi juga menganalisis hubungan antara struktur graf sudoku dan perilaku pencarian solusi secara teoretis, sehingga diperoleh pemahaman matematis mengenai karakteristik penyelesaiannya. Data diperoleh melalui serangkaian percobaan pada berbagai tingkat kesulitan permainan Sudoku, dengan pengukuran meliputi waktu percobaan, jumlah *backtracking* yang dilakukan, dan akurasi hasil. Analisis dilakukan dengan membandingkan kinerja algoritma dalam beragam skenario permainan, serta menilai efisiensinya berdasarkan kompleksitas waktu dan ruang.

## HASIL DAN PEMBAHASAN

Langkah-langkah yang digunakan untuk memecahkan permainan sudoku menggunakan pewarnaan graf dan algoritma backtracking. Langkah yang pertama, setiap kotak matriks sudoku diubah ke dalam bentuk graf sudoku di mana setiap kotak merupakan simpul graf. Langkah yang kedua, simpul-simpul tersebut dengan aturan semua simpul akan bertetangga jika berada dalam baris dan kolom yang sama serta dalam satu kotak matriks  $3 \times 3$  yang sama. Langkah yang ketiga, tentukan warna sebagai penanda pada setiap angka. Langkah yang keempat, warnai simpul-simpul yang sudah terisi angka (yang diberi dari permainan dengan warna yang sesuai berdasarkan langkah nomor 3). Langkah yang kelima, warnai simpul graf yang belum berwarna dengan prinsip pewarnaan graf. Pewarnaan pertama pilih simpul yang paling banyak muncul. Jika bertetangga dengan baris, kolom, dan satu kotak matriks  $3 \times 3$  maka warna tidak boleh sama. Langkah yang keenam, ketika menerapkan prinsip pewarnaan graf terdapat kesalahan, maka akan diterapkan algoritma backtracking dengan mencoba pada kemungkinan yang terdapat pada satu kotak matriks. Langkah yang ketujuh, jika dalam satu kotak matriks terdapat 2 simpul yang dapat diisi dengan warna yang sama. Maka, lakukan dengan algoritma *backtracking*. Langkah yang terakhir, lakukan langkah 4 sampai 7 sampai semua kotak terisi angka.

Sebagai contoh, misalkan diberikan soal Sudoku seperti gambar 1. Langkah pertama, mengubah matriks dan membayangkan sebagai graf dengan aturan yang telah dituliskan di atas. Sebagai contoh, simpul angka 7 akan bertetangga dengan simpul horizontal, vertikal, dan satu kotak matriks  $3 \times 3$  yang diberi warna pada gambar 2.

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

Gambar 1. Contoh Soal Sudoku

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

Gambar 2. Ketetanggaan

Berikutnya, tentukan warna untuk masing-masing angka, misalnya:

**Tabel 1. Pewarnaan pada Sudoku**

Angka	Warna Simpul
1	Merah
2	Kuning
3	Hijau
4	Biru
5	Ungu
6	Oren
7	Pink
8	Abu-abu
9	Toska

Kemudian, warnai sudoku yang ada di papan permainan dengan warna-warna tersebut. Sudoku akan menjadi seperti gambar 3. Selanjutnya, pilih simpul warna pink karena merupakan simpul yang paling banyak muncul pada permainan sudoku. Cari matriks 3×3 yang tidak terdapat simpul berwarna pink serta cari simpul yang tidak bertetangga dengan kolom dan baris simpul berwarna pink, maka simpul yang tidak bertetangga tersebut dapat diberi warna pink (Gambar 4).

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

**Gambar 3. Pewarnaan Simpul**

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

**Gambar 4. Pewaran Awal Simpul Pink**

Jika ada dua kemungkinan seperti gambar di bawah ini, gunakan algoritma *backtracking* dengan mencoba beberapa kemungkinan sampai dapat terpenuhi semua simpul berwarna pink dan tidak ada yang saling bertetangga seperti gambar 5. Gunakan algoritma *backtracking* dengan mencoba beberapa kemungkinan sampai dapat terpenuhi semua simpul berwarna pink dan tidak ada yang saling bertetangga seperti gambar 6.

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

**Gambar 5. Masalah yang Diselesaikan dengan Backtracking**

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

**Gambar 6. Pewarnaan Simpul Pink**

Kemudian, cari simpul warna hijau karena merupakan simpul yang paling banyak muncul setelah simpul warna

pink. Cari matriks 3×3 yang tidak terdapat simpul warna hijau serta cari simpul yang tidak bertetangga dengan kolom dan baris simpul berwarna hijau, maka simpul yang tidak bertetangga tersebut dapat diberi warna hijau (Gambar 7). Cari simpul warna oren karena merupakan simpul yang paling banyak muncul setelah simpul warna hijau. Cari matriks 3×3 yang tidak terdapat simpul warna oren serta cari simpul yang tidak bertetangga dengan kolom dan baris simpul berwarna oren, maka simpul yang tidak bertetangga tersebut dapat diberi warna oren (Gambar 8).

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

**Gambar 7.** Pewarnaan Simpul Hijau

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

**Gambar 8.** Pewarnaan Simpul Oren

Kemudian, cari simpul warna merah karena merupakan simpul yang paling banyak muncul setelah simpul warna oren. Cari matriks 3×3 yang tidak terdapat simpul warna merah serta cari simpul yang tidak bertetangga dengan kolom dan baris simpul berwarna merah, maka simpul yang tidak bertetangga tersebut dapat diberi warna merah (Gambar 9). Selanjutnya, cari simpul warna kuning karena merupakan simpul yang paling banyak muncul setelah simpul warna merah. Cari matriks 3×3 yang tidak terdapat simpul warna kuning serta cari simpul yang tidak bertetangga dengan kolom dan baris simpul berwarna kuning, maka simpul yang tidak bertetangga tersebut dapat diberi warna kuning (Gambar 10).

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

**Gambar 9.** Pewarnaan Simpul Merah

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

**Gambar 10.** Pewarnaan Awal Simpul Kuning

Jika ada dua kemungkinan seperti gambar 11. Gunakan algoritma *backtracking* dengan mencoba beberapa kemungkinan sampai dapat terpenuhi semua simpul berwarna pink dan tidak ada yang saling bertetangga seperti gambar 12.

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

**Gambar 11.** Masalah yang Diselesaikan dengan Backtracking (2)

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

**Gambar 12.** Pewarnaan Simpul Kuning

Kemudian, cari simpul warna ungu karena merupakan simpul yang paling banyak muncul setelah simpul warna kuning. Cari matriks 3×3 yang tidak terdapat simpul warna ungu serta cari simpul yang tidak bertetangga dengan kolom dan baris simpul berwarna ungu, maka simpul yang tidak bertetangga tersebut dapat diberi warna ungu (Gambar 13). Selanjutnya, cari simpul warna biru karena merupakan simpul yang paling banyak muncul setelah simpul warna ungu. Cari matriks 3×3 yang tidak terdapat simpul warna biru serta cari simpul yang tidak bertetangga dengan kolom dan baris simpul berwarna biru, maka simpul yang tidak bertetangga tersebut dapat diberi warna biru (Gambar 14).

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

**Gambar 13.** Pewarnaan Simpul Ungu

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

**Gambar 14.** Pewarnaan Simpul Biru

Kemudian, cari simpul warna abu-abu karena merupakan simpul yang paling banyak muncul setelah simpul warna biru. Cari matriks 3×3 yang tidak terdapat simpul warna abu-abu serta cari simpul yang tidak bertetangga dengan kolom dan baris simpul berwarna abu-abu, maka simpul yang tidak bertetangga tersebut dapat diberi warna abu-abu (Gambar 15). Selanjutnya, cari simpul warna toska karena merupakan simpul yang paling banyak muncul setelah simpul warna abu-abu. Cari matriks 3×3 yang tidak terdapat simpul warna toska serta cari simpul yang tidak bertetangga dengan kolom dan baris simpul berwarna toska, maka simpul yang tidak bertetangga tersebut dapat diberi warna toska (Gambar 16).

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

Gambar 15. Pewarnaan Simpul Abu-abu

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

Gambar 16. Pewarnaan Simpul Toska

Setelah semua simpul diberi warna dengan benar, kembalikan warna menjadi angka.

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

Gambar 17. Hasil Akhir Permainan Sudoku

Teka-teki sudoku sudah berhasil diselesaikan. Dari langkah tersebut, dihasilkan:

#### 1. Struktur Graf Sudoku

Berdasarkan konstruksi graf  $G = (V, E)$  yang telah didefinisikan pada bagian metode, graf sudoku klasik berukuran  $9 \times 9$  memiliki 81 simpul, di mana setiap simpul merepresentasikan satu sel pada grid sudoku. Dua simpul dihubungkan oleh sisi jika dan hanya jika keduanya berada pada baris, kolom, atau subgrid  $3 \times 3$  yang sama. Dengan konstruksi tersebut, graf sudoku merupakan graf tak berarah dengan struktur keteraturan tinggi.

#### 2. Derajat dan Kereguleran Graf Sudoku

##### Teorema 1

Graf sudoku klasik merupakan graf reguler berderajat 20.

##### Pembuktian

Ambil sembarang simpul  $v \in V$ . Simpul tersebut bertetangga dengan:

- 8 simpul lain pada baris yang sama,
- 8 simpul lain pada kolom yang sama,
- 4 simpul tambahan pada subgrid  $3 \times 3$  yang sama yang belum terhitung pada baris dan kolom.

Sehingga total derajat simpul adalah:

$$8 + 8 + 4 = 20.$$

Karena perhitungan ini berlaku untuk setiap simpul pada graf, maka setiap simpul memiliki derajat yang sama, yaitu 20. Dengan demikian graf sudoku merupakan graf reguler berderajat 20.

3. Struktur Klik dan Bilangan Kromatik

Setiap baris, kolom, dan subgrid  $3 \times 3$  membentuk subgraf lengkap berukuran 9.

Teorema 2

Graf sudoku klasik memiliki bilangan kromatik

$$\chi(G) = 9.$$

Pembuktian

Karena setiap baris membentuk klik  $K_9$ , maka graf sudoku memuat subgraf lengkap berukuran 9. Diketahui bahwa jika suatu graf memuat klik berukuran  $r$ , maka:

$$\chi(G) \geq r.$$

Dengan demikian diperoleh:

$$\chi(G) \geq 9.$$

Di sisi lain, solusi sudoku yang valid menggunakan tepat 9 simbol berbeda sehingga tidak ada dua simpul bertetangga memiliki simbol yang sama. Hal ini ekuivalen dengan pewarnaan graf menggunakan 9 warna.

Sehingga:

$$\chi(G) \leq 9.$$

Karena batas bawah dan batas atas sama, maka:

$$\chi(G) = 9.$$

4. Analisis Strategi Pewarnaan

Permasalahan sudoku ekuivalen dengan pencarian pewarnaan graf menggunakan sembilan warna tanpa konflik ketetanggaan.

Strategi pewarnaan yang digunakan bersifat heuristik greedy, yaitu memilih simpul dengan frekuensi kemunculan warna terbanyak untuk mempersempit ruang solusi. Pendekatan ini bertujuan:

- Mengurangi potensi konflik lebih awal,
- Memperkecil cabang eksplorasi pada proses pencarian.

Namun demikian, pewarnaan greedy tidak menjamin solusi global optimal, sehingga diperlukan mekanisme eksplorasi sistematis melalui backtracking.

5. Analisis Kompleksitas Algoritma Backtracking

Teorema 3

Misalkan terdapat  $k$  simpul belum terwarnai pada graf sudoku. Kompleksitas waktu algoritma backtracking dalam kasus terburuk adalah

$$O(9^k).$$

Pembuktian

Setiap simpul yang belum terwarnai memiliki paling banyak 9 kemungkinan warna.

Dalam proses rekursif, algoritma backtracking mencoba setiap kemungkinan warna untuk satu simpul sebelum melanjutkan ke simpul berikutnya. Dalam kasus terburuk, tidak terjadi pruning pada tahap awal sehingga seluruh kombinasi pewarnaan parsial dieksplorasi.

Jumlah maksimum kombinasi adalah:

$$9 \times 9 \times \dots \times 9 = 9^k.$$

Karena kedalaman rekursi maksimum adalah  $k$  dan setiap tingkat memiliki paling banyak 9 percabangan, maka kompleksitas waktu dalam kasus terburuk adalah:



$$O(9^k).$$

#### 6. Implikasi Struktural terhadap Efisiensi Pencarian

Graf sudoku yang reguler berderajat 20 dan memuat banyak klik  $K_9$  menyebabkan kendala pewarnaan sangat ketat. Setiap penetapan warna pada satu simpul secara langsung membatasi 20 simpul lain. Akibatnya, konflik pewarnaan terdeteksi lebih awal dibandingkan graf umum dengan jumlah simpul yang sama. Hal ini menyebabkan proses pruning dalam algoritma backtracking terjadi pada level rekursi awal. Dengan demikian, meskipun kompleksitas teoritis bersifat eksponensial, struktur graf sudoku secara signifikan memperkecil ruang solusi efektif, sehingga pencarian solusi menjadi lebih efisien dalam praktik.

### KESIMPULAN

Penelitian ini berhasil memformulasikan permainan sudoku klasik berukuran  $9 \times 9$  ke dalam model graf tak berarah  $G = (V, E)$  dengan 81 simpul, di mana setiap simpul merepresentasikan satu sel pada grid dan setiap sisi merepresentasikan hubungan ketetanggaan berdasarkan baris, kolom, dan subgrid  $3 \times 3$ . Berdasarkan analisis struktural, diperoleh bahwa graf sudoku merupakan graf reguler berderajat 20 dan memuat klik berukuran 9 pada setiap baris, kolom, dan subgrid.

Dari sifat tersebut dibuktikan bahwa bilangan kromatik graf sudoku memenuhi  $\chi(G) = 9$ , sehingga penyelesaian sudoku ekuivalen dengan permasalahan pewarnaan graf menggunakan sembilan warna berbeda tanpa konflik ketetanggaan. Dengan demikian, pendekatan pewarnaan graf memberikan dasar matematis yang kuat dalam merepresentasikan dan memahami struktur kendala pada sudoku.

Selain itu, analisis terhadap algoritma backtracking menunjukkan bahwa kompleksitas waktu dalam kasus terburuk bersifat eksponensial terhadap jumlah simpul yang belum terwarnai, yaitu  $O(9^k)$ . Namun, struktur graf sudoku yang memiliki derajat tetap dan banyak klik berukuran maksimum menyebabkan proses pruning terjadi lebih awal, sehingga pencarian solusi menjadi lebih efisien dibandingkan graf umum dengan ukuran yang sama.

Secara keseluruhan, integrasi antara analisis struktural graf dan formulasi algoritma backtracking memberikan kerangka matematis yang sistematis dalam menjelaskan karakteristik penyelesaian sudoku, tidak hanya sebagai prosedur komputasional, tetapi sebagai konsekuensi langsung dari sifat-sifat graf yang mendasarinya.

### REKOMENDASI

Berdasarkan hasil analisis struktural graf sudoku dan kajian kompleksitas algoritma backtracking, penelitian selanjutnya dapat diarahkan pada pengembangan pendekatan yang lebih optimal melalui integrasi metode pewarnaan graf dengan teknik constraint propagation, heuristik pemilihan simpul berbasis derajat dinamis, atau algoritma pencarian berbasis pemangkasan (branch and bound). Kajian lebih lanjut juga dapat dilakukan terhadap generalisasi sudoku berukuran  $n^2 \times n^2$  untuk menganalisis perubahan sifat struktural graf, bilangan kromatik, serta implikasinya terhadap kompleksitas penyelesaian.

Selain itu, penelitian lanjutan dapat mengeksplorasi klasifikasi graf sudoku dalam konteks teori kompleksitas komputasi, termasuk keterkaitannya dengan permasalahan NP-Complete, guna memperoleh pemahaman yang lebih mendalam mengenai batas teoretis efisiensi algoritma penyelesaiannya.

### UCAPAN TERIMA KASIH

Penulis menyampaikan ucapan terima kasih kepada semua pihak yang telah memberikan dukungan dan bantuan selama penelitian ini. Terutama kepada Program Studi Tadris Matematika UIN Salatiga yang telah memfasilitasi penulis dalam melaksanakan penelitian.

## DAFTAR PUSTAKA

- Ayusi, R., Triantika, A., & Alaydrus, Z. (2025). *Penerapan Algoritma Bracktracking Dalam Penyelesaian Masalah Sistem Pakar*. 1(1), 57–67.
- Bangun, F. R. (2024). *Menemukan Solusi Dalam Permainan Sudoku*.
- Barahama, R. M., Montolalu, C. E. J. C., & Tumilaar, R. (2021). Eksentrisitas Digraf pada Graf Gir Menggunakan Algoritma Breadth First Search. *D'Cartesian: Jurnal Matematika Dan Aplikasi*, 10(1), 31–36. <https://ejournal.unsrat.ac.id/index.php/decartesian/article/view/32995>
- Buhaerah, Busrah, Z., & Sanjaya, H. (2019). Teori Graf dan Aplikasinya. In *Living Spiritual Quotient*. Danuputri, C., & Santosa, N. (2021). *Aplikasi Pemecahan Soal Sudoku dengan Metode Backtracking*. 6(3), 2622–4615. <http://openjournal.unpam.ac.id/index.php/informatika506>
- Fransiskus Fran, A. H. (2019). Pewarnaan Simpul, Sisi, Wilayah Pada Graf Dan Penerapannya. *Bimaster : Buletin Ilmiah Matematika, Statistika Dan Terapannya*, 8(4), 773–782. <https://doi.org/10.26418/bbmst.v8i4.36037>
- Hanafi, A. A., Hibban, N., Zulfikar, F. M., & Adhinata, F. D. (2021). Penyelesaian Permainan Sudoku Menggunakan Algoritma Backtracking Berbasis Artificial Intelligence. *Journal ICTEE*, 2(2), 50. <https://doi.org/10.33365/jictee.v2i2.1288>
- Martin, T. F., Bandung, I. T., & Bandung, J. G. (2021). *Aplikasi Graf pada Penyelesaian Permainan Sudoku*.
- Meidina, R., & Firdaus, M. (2025). *Implementasi dan Analisis Algoritma Backtracking untuk Penyelesaian Sudoku*. 1(1), 29–35.
- Novianto, S., & Pramadhana, D. (2019). *Penerapan Algorithma Backtracking Dalam Menyelesaikan Permainan Knight Tour*. 205–213.
- Qomaruddin, M., Bismi, W., & Hariyanto, D. (2022). Pewarnaan Graf Pada Peta Provinsi Jawa Barat Menggunakan Algoritma Welch-Powell. *Jurnal Sistem Dan Teknologi Informasi (JustIN)*, 10(2), 258. <https://doi.org/10.26418/justin.v10i2.53829>
- Rahman, F. A., & Anubhakti, D. (2020). Implementasi Algoritma Backtracking Pada Permainan Sudoku. *MEANS (Media Informasi Analisa Dan Sistem)*, 5(1), 67–71. <https://doi.org/10.54367/means.v5i1.711>
- Rifqo, M. H., & Apridiansyah, Y. (2017). Implementasi Algoritma Backtracking Dalam Sistem Informasi Perpustakaan Untuk Pencarian Judul Buku (Studi Kasus Unit Pelayanan Terpadu Perpustakaan Universitas Muhammadiyah Bengkulu). *Pseudocode*, 4(1), 90–96. <https://doi.org/10.33369/pseudocode.4.1.90-96>
- Syam, R., Ihsan, H., & Asman, A. (2020). Aplikasi Pewarnaan Graf dengan Algoritma Recursive Largest First pada Penjadwalan Mata Kuliah. *Journal of Mathematics, Computations, and Statistics*, 2(1), 63. <https://doi.org/10.35580/jmathcos.v2i1.12461>
- Zaturrahmi. (2017). Implementasi Permainan Sudoku Dalam Meningkatkan Pertumbuhan Kognitif Siswa Dalam Lingkungan Belajar. *E-Tech*, 07(00), 1–7. <http://ejournal.unp.ac.id/index.php/e-tech%0ALINGKUNGAN>.