



پروژه رنگ آمیزی گراف با استفاده از جستجو ممنوعه و مدل برنامه
ریزی عدد صحیح

دکتر قنبری، دکتر راهدار

اعضای گروه: سینا ملازاده، الهام خطیبی، سینا مریدا

بهمن ۱۴۰۳

معرفی کلی

مسئله رنگ‌آمیزی گراف یکی از مسائل اساسی و پرکاربرد در نظریه گراف است که به تخصیص رنگ به رأس‌های یک گراف می‌پردازد، به‌گونه‌ای که هیچ دو رأس مجاور (یعنی رأس‌هایی که یال بین آن‌ها وجود دارد) رنگ یکسانی نداشته باشند. هدف اصلی در این مسئله، یافتن حداقل تعداد رنگ‌های ممکن برای رنگ‌آمیزی گراف است که این تعداد به عنوان عدد رنگی گراف شناخته می‌شود. عدد رنگی یکی از ویژگی‌های مهم گراف‌ها به شمار می‌رود و اطلاعات ارزشمندی درباره ساختار و پیچیدگی گراف ارائه می‌دهد.

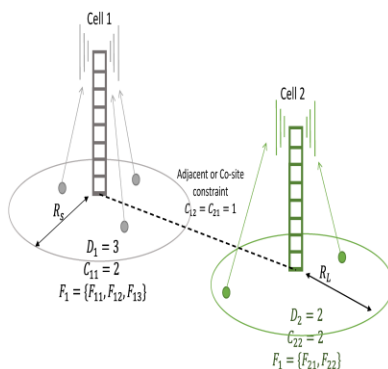
رنگ‌آمیزی گراف علاوه بر اهمیت نظری، کاربردهای گسترده‌ای در مسائل عملی دارد. برای مثال، در زمان‌بندی کلاس‌ها یا امتحانات، می‌توان کلاس‌ها را به‌عنوان رأس‌ها و اشتراک دانشجویان را به‌عنوان یال‌ها مدل‌سازی کرد تا کلاس‌هایی که دانشجویان مشترک دارند، در زمان‌های مختلف برگزار شوند. همچنین، در شبکه‌های بی‌سیم، از رنگ‌آمیزی گراف برای تخصیص بهینه کانال‌های فرکانسی استفاده می‌شود، به‌گونه‌ای که تداخل فرکانسی به حداقل برسد. این پروژه شامل حل مسئله رنگ‌آمیزی گراف است که هدف آن یافتن حداقل تعداد رنگ‌ها برای گراف‌هایی با تراکم و ابعاد مختلف است. در این پروژه از دو روش استفاده شده است:

1. مدل برنامه ریزی عدد صحیح

2. جستجو ممنوعه

کاربردهای مساله

1. تخصیص فرکانس (Frequency Assignment Problem):



در شبکه‌های Wi-Fi غیرمتمرکز (مانند IEEE 802.11)، رنگ‌آمیزی گراف برای انتخاب کانال‌های فرکانسی استفاده می‌شود تا تداخل به حداقل برسد و بازدهی شبکه افزایش یابد. رنگ‌آمیزی طیفی با استفاده از ماتریس تداخل، میزان تداخل کل را محاسبه می‌کند.

2. فشرده‌سازی تصویر (Image Compression):

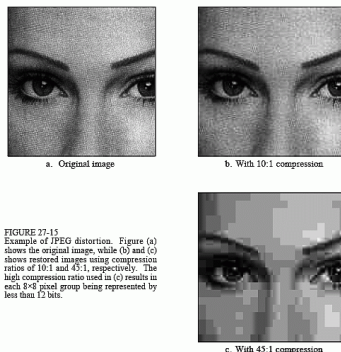
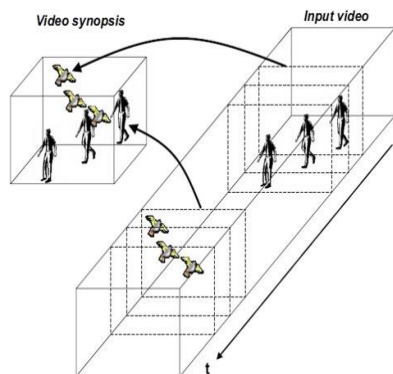


FIGURE 27-15 Example of JPEG distortion. Figure (a) shows the original image, while (b) and (c) shows restored images using compression ratios of 10:1 and 45:1, respectively. The high compression ratio used in (c) results in each 8×8 pixel group being represented by less than 12 bits.

در این کاربرد، گراف رنگ‌آمیزی برای خوشه‌بندی بلوک‌های رنگی در تصاویر دیجیتال استفاده می‌شود. این روش مبتنی بر الگوریتم Welch-Powell و تبدیل Wavelet-SVD است که تعداد رنگ‌ها را به حداقل می‌رساند.



3. خلاصه سازی ویدئو: (Video Synopsis)

برای فشرده سازی هوشمند ویدئوهای نظارتی، از رنگ آمیزی گراف برای تحلیل برخورد بین فریم ها و جداسازی آن ها استفاده می شود. این روش به کاهش زمان بازبینی ویدئوها کمک می کند.

4. تقسیم بندی تومور مغزی:

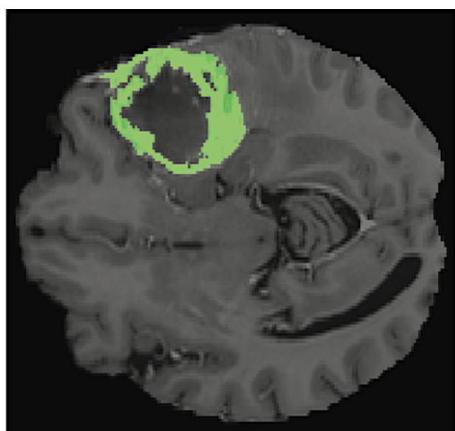


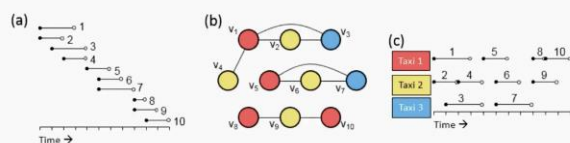
Figure 2: An example of rater's annotation for respective to Figure 1

در تصاویر MRI ، رنگ آمیزی گراف برای شناسایی و جداسازی دقیق تومورهای مغزی به کار می رود. این روش با بهره گیری از گراف های چندبخشی و تحلیل ماتریس های همجواری، داده های تصویری پیچیده را به بخش های معنادار تقسیم کرده و امکان شناسایی دقیق مرزهای تومور را فراهم می کند. به عنوان مثال، این تکنیک می تواند در بهبود دقت در تشخیص و برنامه ریزی درمان های پزشکی مؤثر باشد.

5. برنامه ریزی تاکسی ها: (Taxi Scheduling)

Scheduling Tasks

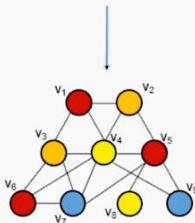
- Scheduling Taxis (and similar)



رنگ آمیزی گراف برای زمان بندی سفرهای تاکسی ها استفاده می شود تا از همپوشانی زمانی جلوگیری شود. در این مدل، هر رأس نماینده یک سفر است و یال بین دو رأس زمانی تعریف می شود که بازه زمانی دو سفر تداخل داشته باشد. هدف این است که هیچ دو سفر همزمان اجرا نشوند و هر رنگ نشان دهنده یک بازه زمانی مجزا باشد.

Lecture Timetabling

Lecture Name	Clashes with...
1) Algebra	2, 3, 4
2) Probability Theory	1, 4, 5
3) Graph Theory	1, 4, 6, 7
4) French Literature	1, 2, 3, 5, 6, 7, 9
5) English Literature	2, 4, 7, 8, 9
6) Welsh Literature	3, 4, 7
7) European Economics	3, 4, 5, 6
8) Algorithms	5
9) C++ Programming	4, 5



6. زمان بندی جدول ها: (Timetabling)

در زمان بندی کلاس ها یا امتحانات، می توان از گراف برای مدل سازی استفاده کرد. هر رأس نشان دهنده یک کلاس یا امتحان است، و یال ها بین دو رأس زمانی ایجاد می شوند که دو کلاس اشتراک دانشجو دارند. سپس با رنگ آمیزی گراف به طوری که کلاس هایی که اشتراک دانشجو دارند رنگ های متفاوتی داشته باشند، می توان اطمینان حاصل کرد که این کلاس ها در زمان های متفاوت برگزار می شوند.

منابع

- [1] Voloshin V, "Graph Coloring: History, results and open problems", Alabama Journal Of Mathematics Spring/Fall 2009
- [2] Z Tuza, Gross, JL, Yellen, J., Eds, Graph Theory Second Edition, CRC Press, 2004
- [3] JM Gimenez-Guzman, I Marsa-Maestre, E de la Hoz, "Channel selection in uncoordinated ieee 802.11 networks using graph coloring" Sensors mdpi.com 2023
- [4] M Maharani, BK Dewi, FA Yulianto, B Purnama, "Digital image compression using graph coloring quantization based on wavelet-SVD" Journal of Physics: Conference Series, 2013
- [5] Y He, C Gao, N Sang, Z Qu, J Han, "Graph coloring based surveillance video synopsis", Neurocomputing, 2017

شرح روش ها

۱. مدل برنامه ریزی عدد صحیح خطی

با استفاده از حل کردن مدل، به تعداد رنگ های لازم برای رنگ آمیزی می رسیم که مدل آن به صورت مقابل است.

متغیر های تصمیم را به این گونه تعریف می کنیم.

For any Graph $G = (V, E)$

$$w_j = \begin{cases} 1 & \text{if any node is assigned to color } j \\ 0 & \text{o. w} \end{cases}$$

$$x_{i,j} = \begin{cases} 1 & \text{if color } j \text{ is assigned to node } i \\ 0 & \text{o. w} \end{cases}$$

در نتیجه تابع هدف ما به این صورت در می آید.

$$\min \sum_{j=1}^{\text{upper bound}} w_j$$

همچنین محدودیت های زیر را داریم:

$$\sum_{j=0}^{\text{upper bound}} x_{i,j} = 1 \quad \forall i \in V$$

که باعث می شود که هر راس دقیقا با یک رنگ، رنگ شود.

$$x_{i,j} \leq w_j \quad \forall i \in V, \forall j \in \text{upper bound}$$

که نشان می دهد که ما راسی را رنگی که استفاده نکرده ایم اختصاص نمی دهیم.

$$x_{i,c} + x_{j,c} \leq 1, \forall (i,j) \text{ where } (i,j) \in E$$

در نهایت شرط بالا شرط همجوار نبودن رئوس هم رنگ را نشان می دهد.

۲. مدل ترکیبیاتی

برای مسئله رنگ آمیزی گراف به روش های مختلفی امکان مدل سازی به صورت ترکیبیاتی موجود می باشد ولی در این گزارش مسئله به صورت زیر مدل شده است.

For any Graph $G = (V, E)$

$$\Omega = \{C \mid \bigcup_{c \in C} c = V, c_1 \cap c_2 = \emptyset \quad \forall c_1, c_2 \in C,$$

$$c \neq \emptyset \quad \forall c \in C, \{i, j\} \not\subset c \quad \forall c \in C \quad \forall (i, j) \in E$$

که به معنی این است که مجموعه شامل تمام افراز هایی از رئوس گراف می باشد که به ازای هر یال، دو سر آن یال در یک افراز وجود نداشته باشد.

۳. جستجو ممنوعه

جستجو ممنوعه یک روش فراابتکاری می باشد که با دریافت یک ساختار همسایگی و یک جواب اولیه سعی در حمله به جواب بهینه سراسری دارد.

روش کلی این روند به این صورت است که یک جواب اولیه را گرفته و ارزش آن را حساب کرده و ذخیره می کند و بعد شروع به پیمایش بر روی ساختار همسایگی می کند و بهترین جواب در همسایگی را پیدا کرده و ارزش آن را در صورت بهتر بودن از جواب های قبلی ذخیره می کند.

جستجو در محور جواب فعلی ادامه پیدا می کند و در هر مرحله جواب قبلی خود را در آرایه ای با ظرفیت محدود ذخیره کرده و خود را از برگشت به آن جواب منع می کند و آرایه از ابتدا شروع به خالی شدن در صورت گذر از ظرفیت می کند.

ساختار همسایگی استفاده شده اینجا به صورت زیر است:

$$S = \{C_1, C_2, \dots, C_n\}$$

$$N(S) = \{S' \mid S' = \{C'_1, C'_2, \dots, C'_n\}, C'_i = \frac{C_i}{\{v\}}, C'_j = C_j \cup \{v\}, C'_k = C_k \text{ for } k \neq i, j\}$$

در این پیاده سازی جواب اولیه برای جستجو ممنوعه اختصاص دادن هر راس به یک رنگ می باشد که الگوریتم با شرط خاتمه بازه زمانی سعی در بهبود جواب دارد و این بازه زمانی تعداد رئوس به ثانیه بود که نتایج آن با سیستم زیر آورده شده است.

Host Name: DESKTOP-NI723MV

OS Name: Microsoft Windows 10 Pro

System Manufacturer: LENOVO

System Type: x64-based PC

Processor(s): 1 Processor(s) Installed.

: [01] Intel64 Family 6 Model 186 Stepping 2 GenuineIntel ~2100 Mhz

Total Physical Memory: 7,885 MB

Virtual Memory: Max Size: 9,101 MB

ساختار پوشه‌ها

درون برنامه پروژه پوشه ای به نام انگلیسی Plots قرار دارد که در آن نتایج حاصل از رنگ آمیزی روی گراف ها با ابعاد و چگالی های مختلف ذخیره شده است:

درون نام هر پوشه داخل پوشه Plots، شما با دو عدد مواجه می شوید که به ترتیب تعداد راس ها و چگالی هر گراف است و درون آن ها سه تصویر و یک فایل متنی قرار دارد.

1. گراف قبل از رنگ آمیزی (تصویر ۱)
2. گراف بعد از رنگ آمیزی با جستجو ممنوعه (تصویر ۲)
3. گراف بعد از رنگ آمیزی با مدل برنامه ریزی عدد صحیح (تصویر ۳)
4. نقشه رنگ گراف، تعداد رنگ های استفاده شده و زمان صرف شده برای هر الگوریتم.

در ادامه پرونده شما با سه تصویر دیگر رو به رو خواهید شد که به ترتیب

1. مقایسه تعداد رنگ های استفاده شده برای گراف های متفاوت (تصویر ۴)
2. زمان صرف شده توسط هر کدام از روش های رنگ آمیزی (تصویر ۵)
3. پدهستان تعداد رنگ استفاده شده و زمان صرف شده (تصویر ۶)

پرونده های دیگر برای حل مسئله به صورت زیر آمده است.

1. پرونده اصلی (main.py): این پرونده نقش حیاتی در روند انجام پروژه را انجام می دهد که به ازای هر گراف، آن را به فرم ماتریس از پرونده آزمایش گرفته و یک کران بالا روی آن حساب کرده (یک به اضافه بیشترین درجه راس) و زمان احتساب آن را به هر روشی که قصد استفاده از آن را دارد می دهد. در ادامه با فراخوانی توابع از پرونده های دیگر گراف را رنگ آمیزی کرده و نتایج نهایی رو روی نمودار های مربوطه قرار می دهد.
2. پرونده نمودار (plots.py): این پرونده با گرفتن یک نقشه از رنگ آمیزی با استفاده از کتابخانه های مختلف گراف را با رنگ های آن رسم می کند.
3. پرونده مدل (model.py): در این پرونده ما با استفاده از کران بالا و مدل برنامه ریزی عدد صحیح ذکر شده گراف را رنگ آمیزی کرده و نتایج حاصل را اعم از نقشه رنگی، زمان صرف شده و تعداد رنگ را بر می گرداند.

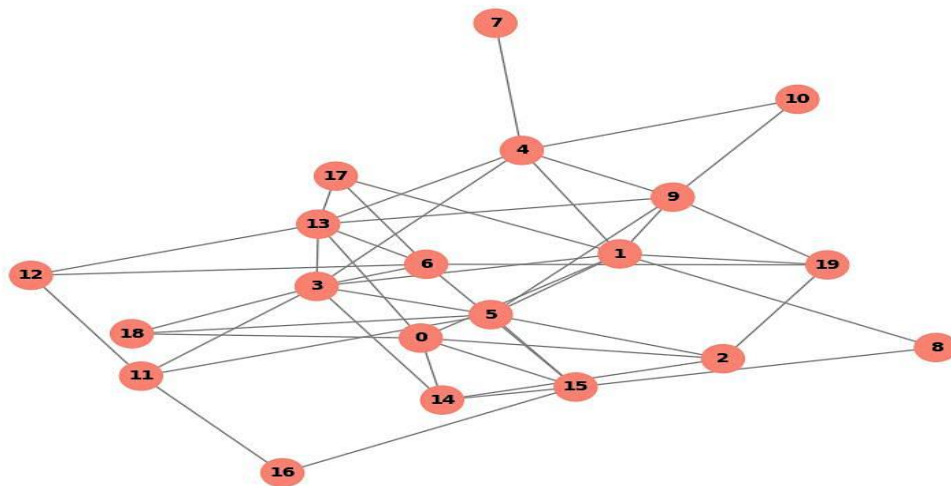
4. پرونده جستجو ممنوعه (tabu.py): در ابتدا به تعداد تمام راس ها رنگ داریم و هر راس به یک رنگ اختصاص داده می شود و مقدار ارزش آن ها و خود جواب ذخیره شده و تا زمانی که الگوریتم به شرط خاتمه خود که حل مسئله در زمان خود است نرسد، یک راس از یک رنگ برداشته و آن را جای دیگری قرار می دهد و ارزش هر جواب تعداد رنگ هایی است که حداقل یک راس به آن ها اختصاص داده شده است.
5. پرونده آزمایش (testCase.py): در این پرونده با دریافت تعداد راس ها و چگالی یک گراف ساخته و آن را به فرم ماتریسی برمی گرداند.

نمونه بزرگ برای جستجو ممنوعه

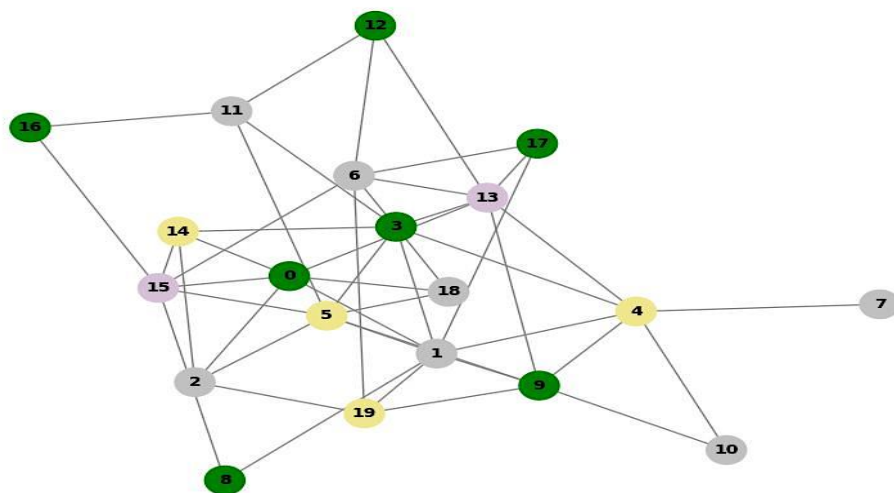
به علت زیاد بودن زمان صرف شده توسط مدل عدد صحیح برای رنگ آمیزی گراف امکان اجرا نمونه های بزرگ با این مدل نیست اما به دلیل وجود الگوریتم جستجو ممنوعه این کار امکان پذیر است پس در این گزارش این الگوریتم را روی گرافی با صد راس و چگالی چهار دهم اجرا کرده و ساختار این روند به صورت مقابل است.

1. گراف قبل از رنگ آمیزی (تصویر ۷)
2. گراف بعد از رنگ آمیزی با جستجو ممنوعه (تصویر ۸)
3. نمودار زمان و رنگ استفاده شده توسط جستجو ممنوعه (تصویر ۹)

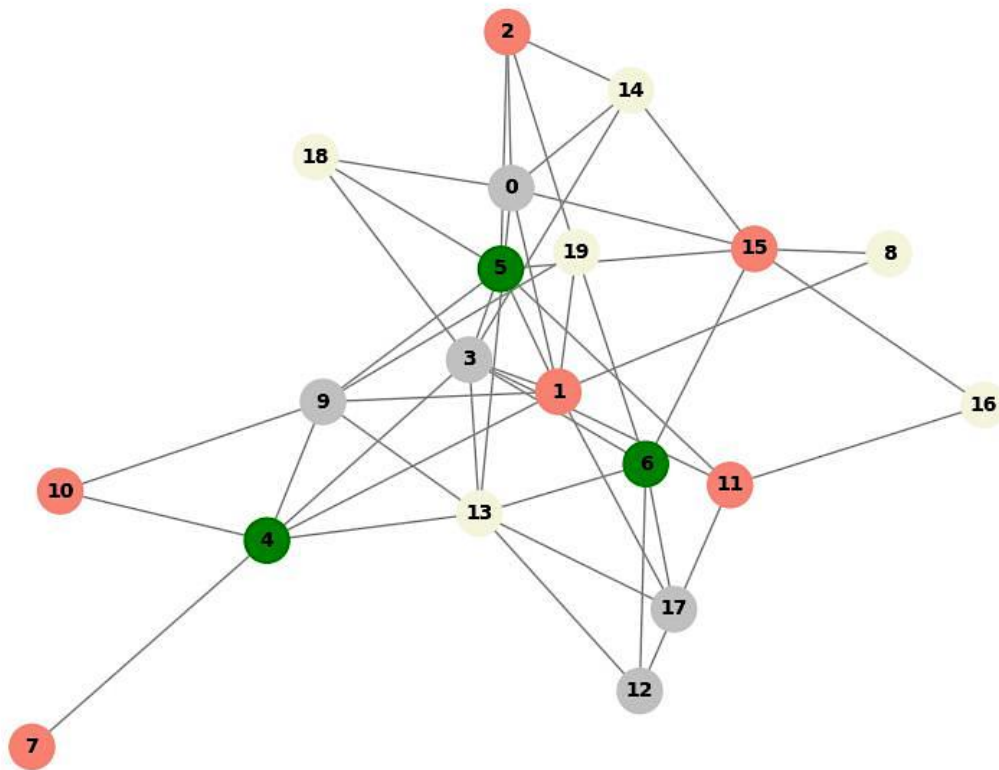
تصاویر



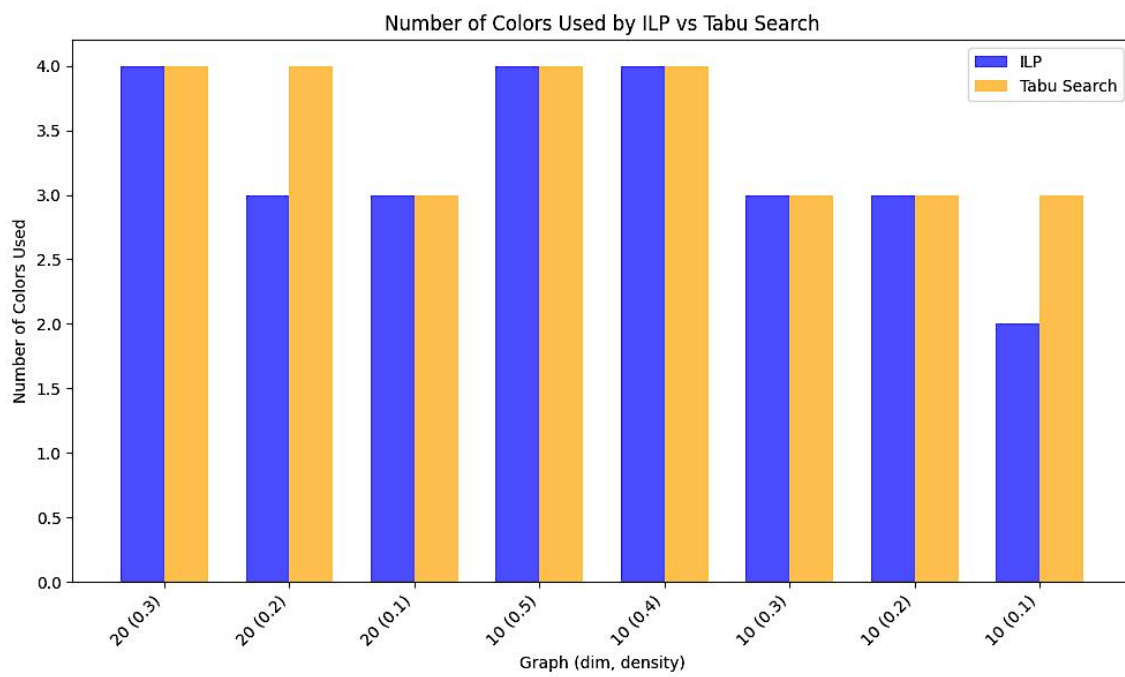
تصویر (۱)



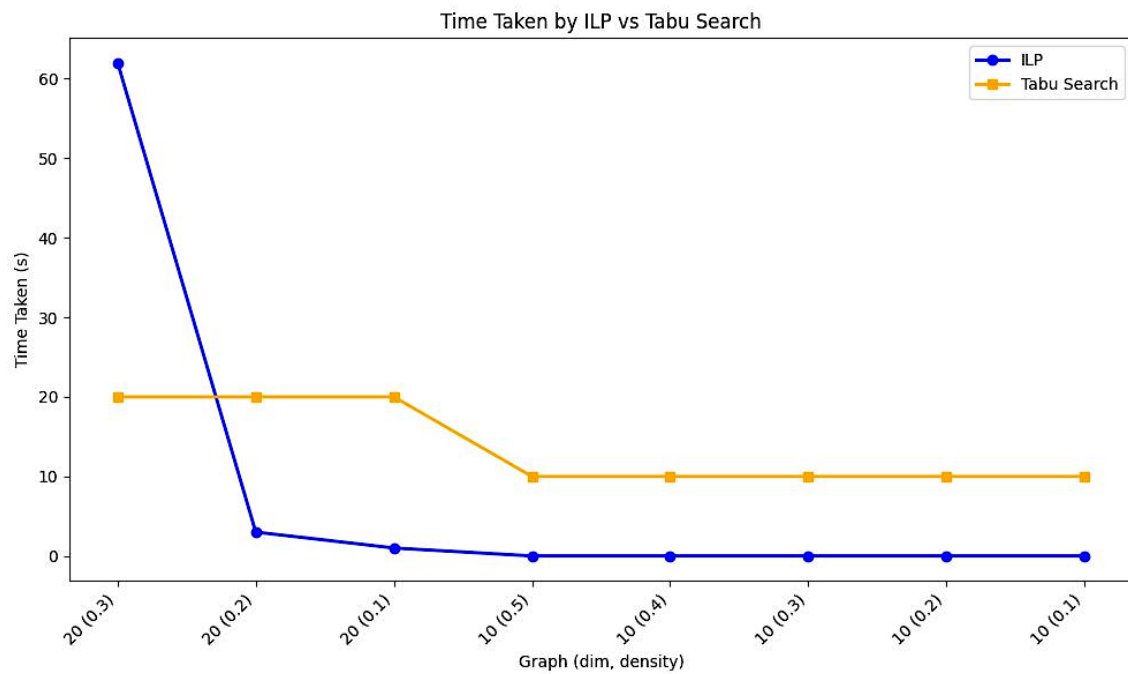
تصویر (۲)



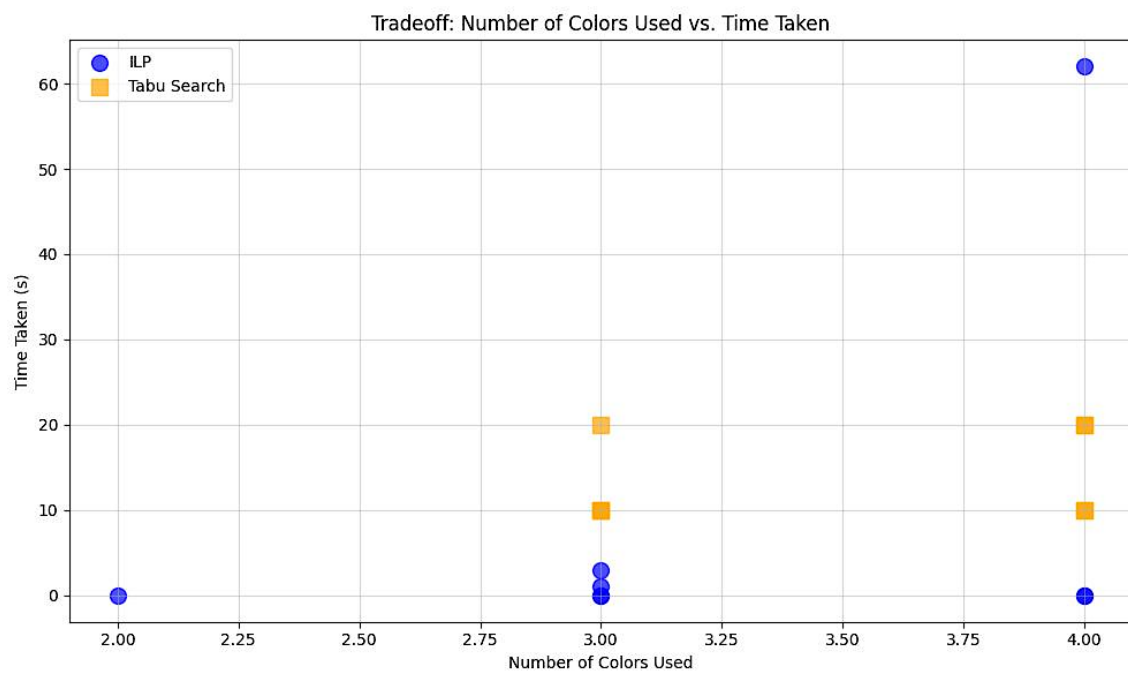
تصویر (۳)



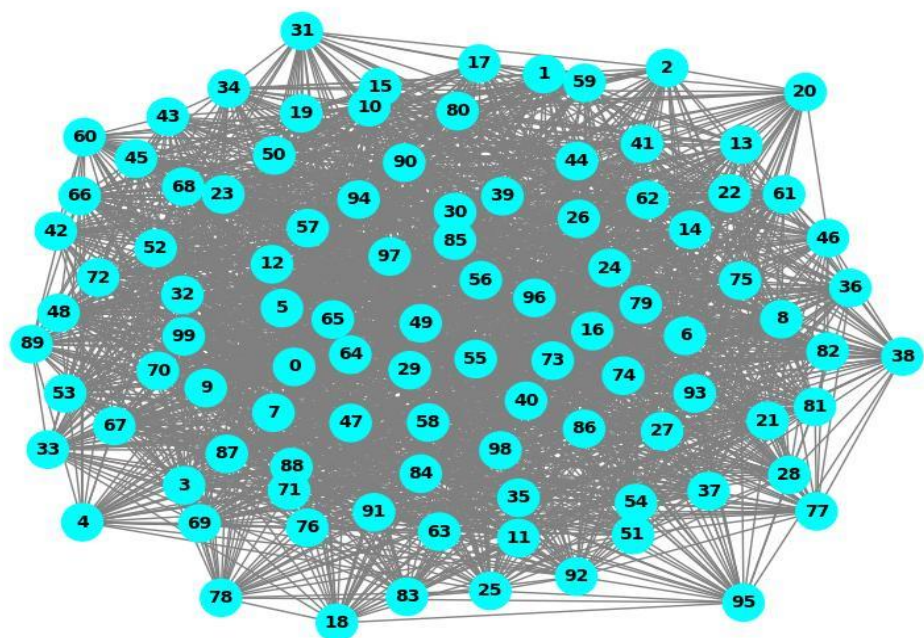
تصویر (۴)



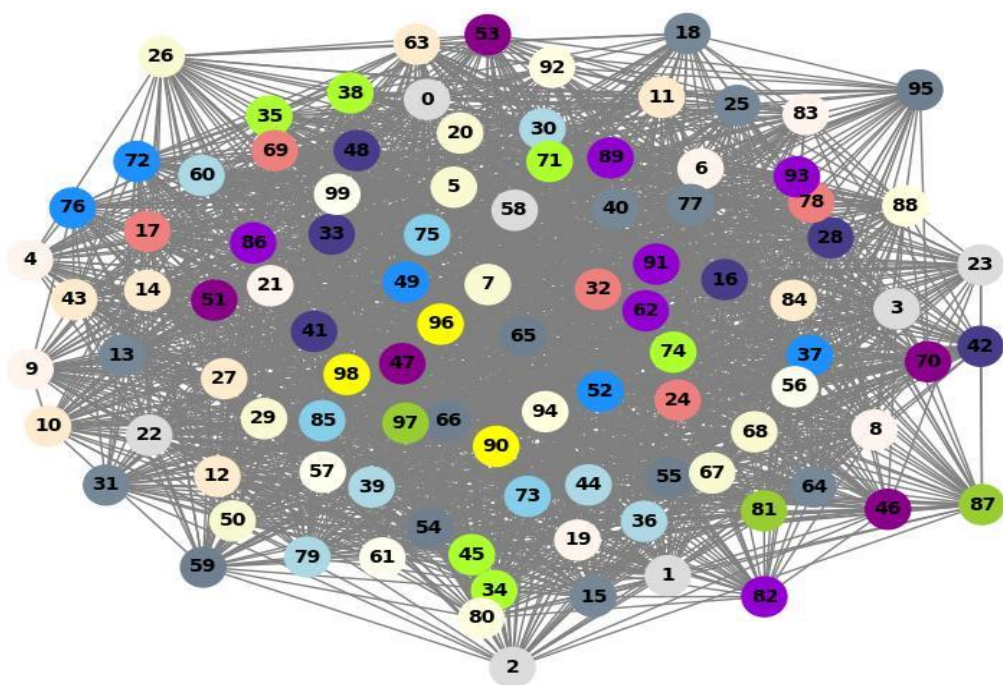
تصویر (۵)



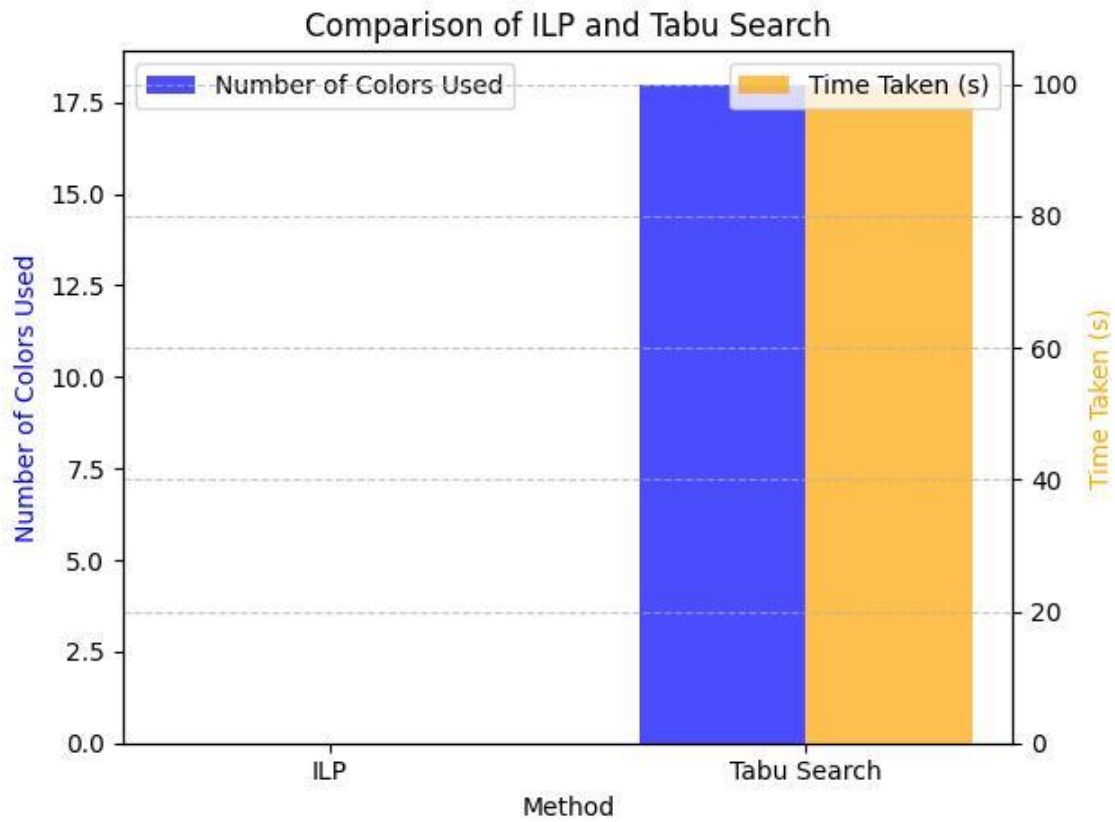
تصویر (۶)



تصویر (۷)



تصویر (۸)



تصویر (۹)