ZESZYTY NAUKOWE POLITECHNIKI ŚLASKIEJ 1994
Seria: AUTOMATYKA z.114 Nr kol. 1250

Marek KUBALE
Politechnika Gdańska

PROBLEM KOLOROWANIA WIERZCHOŁKÓW GRAFÓW. PRZEGLĄD ALGORYTMÓW I ZASTOSOWAŃ

Streszczenie: Problemy kolorowania grafów należą do najtrudniejszych zagadnień kombinatorycznych w sensie złożoności obliczeniowej. Ostatnio pokazano, że skonstruowanie względnie przybliżonego algorytmu wielomianowego dla kolorowania wierzchołków grafu jest problemem NP-trudnym. W niniejszej pracy dokonano przeglądu oszacowań liczby chromatycznej grafu, przeglądu zastosowań problemu kolorowania oraz przeglądu przybliżonych algorytmów kolorowania wierzchołków ze szczególnym uwzględnieniem ich złożoności obliczeniowej, dokładności i struktury trudnych do kolorowania grafów.

PROBLEM OF COLORING THE VERTICES OF A GRAPH. THE SURVEY OF ALGORITHMS AND APPLICATIONS

Summary: Graph coloring problems belong to the hardest combinatorial problems with respect to the computational complexity. Recently, it has been shown that designing of a relative approximation polynomial algorithm for coloring graph vertices is NP-hard. In the paper we review some bounds on the chromatic number of a graph, practical applications of vertex coloring, and approximate algorithms for graph coloring with particular reference to their computational complexity and a structure of hard to color graphs.

DAS PROBLEM DER GRAPHENFÄRBUNG. ÜBERSICHT DER ALGORITHMEN UND ANWENDUNGEN

Zusammenfassung: Wenn man als Kriterium die rechnerische Komplexität nimmt, gehört die Graphenfärbung zu den schwierigsten kombinatorischen Problemen. Neuerdings hat sich erwiesen, daß die Entwicklung, der relativ angenäherten Polynomial-Algorithmen, ein NP-schwieriges Problem ist. Diese Arbeit ist eine Übersicht der Wertungen der chromatischen Zahl, der praktischen Anwendungen der Knotenpunktfärbung und der angenäherten Algorithmen für Graphenfärbung bei Berücksichtigung ihrer rechnerischen Komplexität und der Strukturen welche schwer zu färben sind.

1. Wprowadzenie

Problemy kolorowania grafów należą do najtrudniejszych problemów kombinatorycznych. Ostatnio pokazano na przykład, że skonstruowanie względnie przybliżonego algorytmu wielomianowego dla kolorowania wierzchołków grafu jest problemem NP-trudnym [22]. Ogólnie w problemie kolorowania grafów wyróżniamy dwa zadania: kolorowanie wierzchołków i kolorowanie krawędzi. Pomimo że oba zadania są NP-zupełne, kolorowanie wierzchołków jest trudniejsze z punktu widzenia możliwości skonstruowania efektywnych algorytmów przybliżonych. Oba problemy mają też wiele zastosowań praktycznych w postaci klasycznej i rozszerzonej.

Niniejszy tekst jest pierwszym z dwóch artykułów przeglądowych poświęconych metodom kolorowania grafów. Rozważymy tutaj zagadnienie kolorowania wierzchołków grafu na tle zastosowań w wybranych zagadnieniach elektroniki, telekomunikacji, automatyki i informatyki. W szczególności dokonamy przeglądu dolnych i górnych oszacowań liczby chromatycznej grafu, przeglądu wielomianowo kolorowalnych klas grafów oraz przeglądu praktycznych zastosowań problemu kolorowania wierzchołków grafu. Następnie zaprezentujemy niektóre przybliżone algorytmy kolorowania wierzchołków ze szczególnym uwzględnieniem ich złożoności obliczeniowej, dokładności i struktury trudnych do kolorowania grafów.

2. Pojęcie liczby chromatycznej

Niech będzie dany graf $G = (V,E)$ bez pętli i krawędzi wielokrotnych. Zbiór V mocy $|V| = n$ nazywamy *zbiorem wierzchołków*, zaś zbiór E mocy $|E| = m$ nazywamy *zbiorem krawędzi*. Mówimy, że G jest *k-barwny*, jeśli jego wierzchołki można pokolorować k różnymi barwami w ten sposób, że żadne wierzchołki sąsiednie nie są pokolorowane tą samą barwą. Mówimy wtedy również, że istnieje *k-pokolorowanie* grafu G. Najmniejszą liczbę k, dla której G jest k-barwny, nazywamy *liczbą chromatyczną* grafu G i oznaczamy przez $\chi(G)$, zaś graf ten nazywamy *k-chromatycznym*. Zatem k-pokolorowanie grafu jest podziałem zbioru V na k podzbiorów niezależnych.

Istnieje wiele metod kolorowania wierzchołków dowolnego grafu, zarówno dokładnych, jak i przybliżonych. Niech $A(G)$ oznacza liczbę kolorów użytych do pokolorowania grafu G metodą A. Z każdą taką metodą kolorowania związać można *funkcję dobroci* A: $N \to R$, gdzie $N = \{1, 2, ...\}$ zdefiniowaną następująco:

$$A(n) = \max \{A(G)/\chi(G): G \text{ ma najwyżej } n \text{ wierzchołków}\}.$$

Funkcja ta charakteryzuje jakość rozwiązań generowanych przez algorytm A w najgorszym przypadku danych. Zauważmy, że dla każdego algorytmu kolorowania A mamy $A(n) \leq n$, natomiast dla każdego algorytmu dokładnego $A(n) = 1$.

Dla danego algorytmu A, *najmniejszy trudny* do kolorowania graf definiuje się jako najmniejszy w sensie liczby wierzchołków graf G, taki że pewna implementacja algorytmu A daje wynik $A(G) > \chi(G)$. Jeśli jest więcej takich grafów n-wierzchołkowych, wybiera się ten, który ma najmniejszą liczbę krawędzi. Natomiast *najmniejszym bardzo trudnym* do kolorowania grafem jest taki najmniejszy graf G, dla którego każda implementacja algorytmu A daje wynik $A(G) > \chi(G)$.

Jest jasne, że $\chi(K_n) = n$, gdzie K_n jest grafem pełnym rzędu n i w związku z tym można łatwo zbudować grafy o dowolnie dużej liczbie chromatycznej. Z drugiej strony $\chi(G) = 1$ wtedy i tylko wtedy, gdy G jest grafem pustym oraz $\chi(G) = 2$ wtedy i tylko wtedy, gdy G jest niepustym grafem dwudzielnym (czyli nie zawiera cyklu nieparzystego). Nie wiadomo, przy jakich warunkach graf G jest 3-chromatyczny, chociaż łatwo podać przykłady takich grafów (np. cykliczne o nieparzystej liczbie wierzchołków). Ogólnie nie jest znany żaden prosty wzór na liczbę chromatyczną dowolnego grafu i musimy zadowolić się jej oszacowaniami, zarówno z dołu, jak i z góry.

Dolne oszacowanie liczby chromatycznej wyznaczone jest przez rozmiar największej kliki grafu G, który oznaczać będziemy przez ω. Ponieważ w dowolnym pokolorowaniu grafu G wszystkie wierzchołki kliki muszą otrzymać różne kolory, więc

$$\omega \le \chi(G) \tag{1}$$

Niestety, nie jest znana żadna efektywna metoda wyznaczania liczby klikowej ω i panuje głęboki pesymizm odnośnie tego, czy wartość ω może być policzona w czasie wielomianowym. Również dokładność tego oszacowania nie jest zadowalająca w najgorszym przypadku. Istnieją bowiem grafy $M_1, M_2, \ldots$, takie że $\chi(M_i) = i$, mimo iż graf M_i nie zawiera trójkąta, czyli $\omega(M_i) \le 2$, dla $i = 1,2,\ldots$.

Innym oszacowaniem dolnym jest

$$\frac{n^2}{n^2 - 2m} \le \chi(G), \tag{2}$$

które jest łatwo obliczalne dla każdego grafu G. Jednakże jego dokładność jest gorsza niż (1), gdyż $n^2/(n^2-2m) \le \omega$.

Nieco lepsza sytuacja panuje w zakresie górnych oszacowań liczby chromatycznej. Jeżeli Δ oznacza maksymalny stopień wierzchołka grafu G, to G jest $(\Delta+1)$-barwny, czyli

$$\chi(G) \le \Delta+1. \tag{3}$$

Poprawność tego oszacowania można łatwo wykazać przez indukcję względem liczby wierzchołków grafu.

Brooks [5] udowodnił, że istnieją tylko dwie klasy grafów, dla których zachodzi równość w oszacowaniu (3). Są to: grafy cykliczne C_{2k+1}, dla których $\chi(C_{2k+1}) = 3$ i $\Delta(C_{2k+1}) = 2$ oraz grafy pełne K_n, dla których $\chi(K_n) = n$ i $\Delta(K_n) = n-1$. Zatem, jeśli $G \neq K_n$ i $\Delta \ge 3$, to

$$\chi(G) \le \Delta. \tag{4}$$

Najgorszym z punktu widzenia oszacowań (3) i (4) typem grafów są gwiazdy $K_{1,k}$, dla których $\Delta(K_{1,k}) = k$. Zatem różnica $\Delta(K_{1,k}) - \chi(K_{1,k}) = k-2$ może być dowolnie duża, gdy k rośnie do nieskończoności. Oba oszacowania mogą być obliczone w czasie $O(m)$.

Innym łatwo policzalnym oszacowaniem z góry jest

$$\chi(G) \le \sqrt{2m}+1. \tag{5}$$

Oszacowanie to jest bardzo kiepskie dla grafów dwudzielnych $K_{k,k}$, które mają k^2 krawędzi. Zatem dla tych grafów różnica $\sqrt{2k^2}+1-\chi(G)=\sqrt{2}k-1$ może być dowolnie duża, gdy $k\to\infty$.

Innym prostym oszacowaniem górnym jest

$$\chi(G) \leq \lambda+1, \tag{6}$$

gdzie λ jest długością najdłuższej drogi prostej w grafie G. Niestety, znalezienie najdłuższej drogi w G jest problemem NP-trudnym. Co więcej, oszacowanie to nie jest precyzyjne, co widać na przykładzie ścieżek P_n, dla których $\lambda(P_n) = n-1$.

3. Złożoność obliczeniowa kolorowania

Naczelnym pytaniem w dziedzinie kolorowania grafów jest pytanie o złożoność obliczeniową problemu optymalnego kolorowania dowolnego grafu G. Niestety, nie jest znany rząd złożoności obliczeniowej, a jedynie dolne i górne oszacowanie liczby działań $\phi(m,n)$ potrzebnych do rozwiązania tego problemu. Mianowicie,

$$\Omega(m+n) \leq \phi(m,n) \leq O(mn2.45^n). \tag{7}$$

Oszacowanie dolne wynika stąd, że z jednej strony każda krawędź jest tu istotna i musi być sprawdzona co najmniej raz, z drugiej zaś strony każdy wierzchołek musi mieć nadany jakiś kolor. Oszacowanie górne wynika z analizy złożoności pewnego algorytmu optymalnego kolorowania grafów dokonanej przez Lawlera [21].

Oszacowanie (7) jest oczywiście bardzo nieprecyzyjne. Wynika to stąd, że problem określenia wartości liczby chromatycznej w przypadku ogólnym jest NP-zupełny [19]. Co więcej, zagadnienie to pozostaje trudne do rozwiązania nawet wówczas, gdy G jest grafem należącym do jednej z poniższych rodzin:

1. Grafy planarne.
2. Grafy krawędziowe.
3. Grafy łukowe.
4. Grafy o ograniczonym stopniu.
5. Grafy o ograniczonej grubości.
6. Grafy o ograniczonym rodzaju.
7. Grafy bezpazurowe.

Z drugiej strony problem kolorowania wierzchołków można rozwiązać w czasie wielomianowym, gdy G jest grafem należącym do jednej z poniższych rodzin:

1. Grafy zewnętrznie planarne.
2. Grafy dwudzielne.
3. Grafy kubiczne.
4. Grafy doskonałe.
5. Grafy cięciwowe.
6. Grafy przedziałowe.

7. Grafy szeregowo-równoległe.

8. Grafy permutacyjne.

Więcej szczegółów na ten temat wraz z odpowiednią bibliografią można znaleźć w [18].

4. Przegląd zastosowań

Jak wspomniano na wstępie, problem kolorowania wierzchołków grafu ma wiele zastosowań praktycznych w nauce i technice, zarówno w postaci czystej, jak i rozszerzonej, tzn. z uwzględnieniem dodatkowych kryteriów i ograniczeń. Jednakże musimy pamiętać, że im bardziej dany problem techniczny jest ogólny, tym słabszy pozostaje jego aspekt chromatyczny. Poniżej podajemy listę udokumentowanych zastosowań kolorowania wierzchołkowego w rozbiciu na odpowiednie grupy tematyczne z dziedziny elektroniki i informatyki.

A. Diagnostyka elektroniczna.

1. Minimalizacja liczby testów lokalizujących uszkodzenia układu logicznego [1].
2. Testowanie poprawności połączeń drukowanych na płytce obwodu LSI [12].

B. Technologia elektroniczna.

3. Podział układu elektronicznego na warstwy [2].
4. Pewne aspekty tzw. umieszczania grafów w książkach [9].
5. Przydział kanałów dla połączeń drukowanych na płytce obwodu LSI [14].

C. Telekomunikacja.

6. Projektowanie kodów wykrywających błędy w systemach transmisji danych [7].
7. Przydział częstotliwości roboczych dla telefonii komórkowej [25].

D. Informatyka.

8. Minimalizacja tablic roboczych dla kompilatorów [11].
9. Przydział rejestrów maszyny cyfrowej [8].

E. Rachunek macierzowy.

10. Szacowanie rozrzedzonych macierzy Jacobiego [10].
11. Przybliżanie rozrzedzonych macierzy Hessego [24].
12. Oszczędność pamięci dla macierzy rozrzedzonych [28].

F. Szeregowanie zadań.

13. Układanie rozkładów zajęć dydaktycznych [29].
14. Przydział pracowników do prac o ustalonym harmonogramie [6].

Oczywiście, powyższe konkretne zastosowania problemu kolorowania wierzchołków nie wyczerpują wszystkich przypadków, gdzie stosuje się algorytmy kolorowania grafów (pomijamy tutaj np. zastosowania ekonomiczne). Najbardziej popularne spośród tych algorytmów prezentujemy w następnym punkcie.

5. Przegląd algorytmów przybliżonych

W praktyce, gdy trzeba kolorować grafy mające kilkaset wierzchołków, zmuszeni jesteśmy stosować algorytmy przybliżone. Dlatego jest niezwykle istotne gruntowne poznanie własności takich algorytmów, a zwłaszcza złożoności obliczeniowej, dobroci kolorowania i struktury trudnych do kolorowania grafów.

Istnieje kilkadziesiąt opublikowanych metod przybliżonego kolorowania wierzchołków grafu. Można je podzielić na metody *liniowe*, tj. takie, które mają liniową funkcję dobroci, oraz *subliniowe*, tj. takie, których funkcja dobroci rośnie wolniej niż liniowo. Ponieważ algorytmy liniowe są zwykle prostsze w implementacji, w dalszym ciągu ograniczymy się do omówienia takich właśnie metod. Mimo ograniczenia zakresu naszych zainteresowań nie jesteśmy w stanie dokonać pełnego przeglądu algorytmów liniowych. Dlatego skoncentrujemy się na najbardziej klasycznych algorytmach o liniowej funkcji dobroci, tj. na algorytmach sekwencyjnych.

Algorytm *sekwencyjny* zdefiniowany jest następująco. Niech wierzchołki grafu G będą ustawione w ciąg $v_1,...,v_n$. Wówczas pokoloruj v_1 kolorem 1, a każdy następny wierzchołek v_i kolorem o możliwie najniższym numerze. Łatwo zauważyć, że pokolorowania otrzymywane tą drogą w istotny sposób zależą od przyjętego uszeregowania wierzchołków. Rzeczywiście, jeśli c_0 jest optymalnym pokolorowaniem grafu G, to kolorowanie wierzchołków ustawionych w kolejności niemalejących wartości $c_0(v_i)$ prowadzi do pokolorowania optymalnego. A więc wystarczy znać właściwą kolejność kolorowania wierzchołków. Lecz znalezienie takiej kolejności w zbiorze $n!$ możliwych nie jest łatwe.

Z algorytmem sekwencyjnym związane jest następujące oszacowanie liczby chromatycznej grafu. Niech $S(G; v_1,..,v_n)$ będzie oszacowaniem liczby kolorów użytych przez algorytm sekwencyjny S zgodnie z uporządkowaniem $v_1,...,v_n$. Wówczas

$$\chi(G) \leq S(G; v_1,..,v_n) = \max_{i \leq n} \min\{i, \rho(v_i)+1\}, \tag{8}$$

gdzie $\rho(v_i)$ jest stopniem wierzchołka v_i, $i=1,...,n$. Co więcej, Grimmet i McDiarmid [13] pokazali, że w przypadku przeciętnym algorytm sekwencyjny daje pokolorowania prawie na pewno nie odbiegające w dwójnasób od liczby chromatycznej.

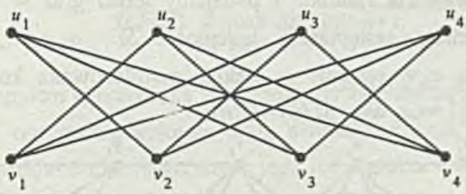
Istnieją trzy podstawowe sposoby porządkowania wierzchołków grafu, będące źródłem trzech różnych algorytmów sekwencyjnych: *LF*, *SL* i *SLF*. Dwa pierwsze są *statyczne* w tym sensie, że w drugiej fazie algorytmu początkowa kolejność wierzchołków nie ulega zmianie. Trzecia metoda jest *dynamiczna*, bo wybór następnego wierzchołka zależy od kolorów przydzielonych jego poprzednikom.

5.1. Metoda LF

W metodzie *pierwszy-największy*, zwanej symbolicznie *LF* (ang. *Largest-First*), wierzchołki są najpierw porządkowane w kolejności nierosnących stopni, a następnie odbywa się właściwe kolorowanie sekwencyjne. Algorytm ten został po raz pierwszy podany przez Welsha i Powella [29] w nieco odmiennej, lecz równoważnej postaci. Jego złożoność jest proporcjonalna do rozmiaru grafu, czyli $O(m+n)$. Można pokazać, że dla każdego grafu G

$$LF(G) \leq \max_{i \leq n} \min \{i, \rho(v_i)+1\}. \qquad (9)$$

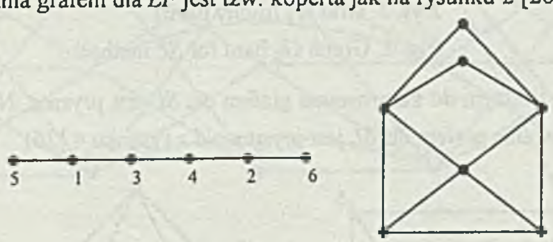
Algorytm pierwszy-największy optymalnie koloruje cykle nieparzyste, gwiazdy, a ogólnie pełne grafy k-dzielne. Jednakże czasami nie jest on w stanie optymalnie pomalować nawet ścieżek. Rodzinę najgorszych do kolorowania grafów dla LF podał Johnson [17]. Jest to rodzina pełnych regularnych grafów dwudzielnych $G_k = (V_k, E_k)$, z których wyjęto k naprzeciwległych krawędzi. Na rysunku 1 podano czwarty z kolei graf Johnsona.



Rys. 1. Graf G_4 trudny dla LF

Fig. 1. Graph G_4 hard for LF method

Jeżeli wierzchołki kolorowane są w porządku $u_1, v_1, \ldots, u_k, v_k$, to każda para malowana jest nowym kolorem. Zatem $LF(G_k) = k = n/2$. Skoro $|V_k| = 2k$ i $\chi(G_k) = 2$, to $LF(n) = O(n)$. Najmniejszym trudnym do kolorowania grafem dla LF jest ścieżka P_6. Najmniejszym bardzo trudnym do kolorowania grafem dla LF jest tzw. koperta jak na rysunku 2 [26].



Rys. 2. Najmniejsze trudne grafy dla LF

Fig. 2. The smallest hard graphs for LF method

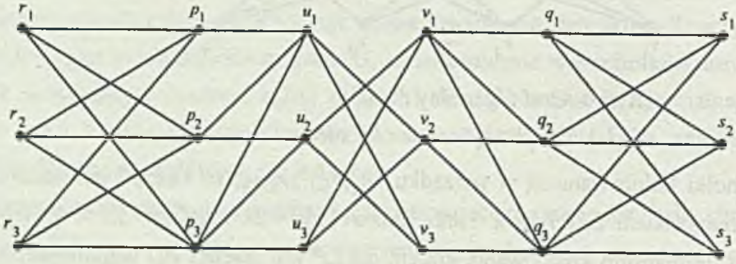
5.2. Metoda SL

W metodzie ostatni-najmniejszy, podanej przez Matulę i inn. [23] i nazwanej symbolicznie SL (ang. *Smallest-Last*), wierzchołki są porządkowane $v_1, \ldots, v_n$ w taki sposób, że dla każdego $i = 1, \ldots, n$ wierzchołek v_i ma minimalny stopień w podgrafie grafu G indukowanym przez wierzchołki $v_1,...,v_i$. Porządkowanie wierzchołków można zinterpretować jako proces redukcji grafu polegającej na sukcesywnym usuwaniu wierzchołków o aktualnie najmniejszym stopniu. Algorytm *SL* można zrealizować w czasie $O(m+n)$. Można też pokazać, że

$$SL(G) \le \max_{i \le n} \min_{j \le i} \{i, \rho_i(v_j)+1\}, \tag{10}$$

gdzie $\rho_i(v_j)$ jest stopniem wierzchołka v_j w podgrafie indukowanym sekwencją *SL* ograniczoną do i pierwszych wierzchołków grafu G.

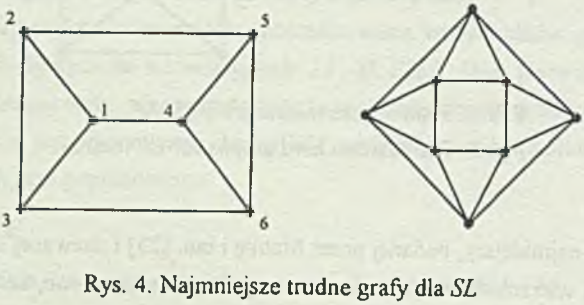
Algorytm ostatni-najmniejszy optymalnie koloruje drzewa, cykle i grafy unicykliczne, koła, grafy Mycielskiego oraz grafy złożone z niezależnych ścieżek łączących dwa ustalone wierzchołki. Ponadto *SL* koloruje prawie optymalnie grafy planarne. Jednakże ma on duże kłopoty już z grafami dwudzielnymi. Rodzinę trudnych do kolorowania grafów dwudzielnych podali Coleman i More [10]. Na rysunku 3 podajemy trzeci graf w tej rodzinie. Jeżeli wierzchołki malujemy według następującej kolejności *SL*: $q_1,s_1,...,q_k,s_k,p_1,r_1,...,p_k,r_k,$ $u_1,v_1,..,u_k,v_k$, to każda para u_i,v_i zostanie w efekcie pokolorowana kolorem i+1. Zatem $SL(G_k)=k+1$ i ponieważ $|V_k| = 6k$, więc $SL(n)=O(n)$.



Rys. 3. Graf G_3 trudny dla *SL*

Fig. 3. Graph G_3 hard for *SL* method

Najmniejszym trudnym do kolorowania grafem dla *SL* jest pryzma. Najmniejszym bardzo trudnym do kolorowania grafem dla *SL* jest pryzmatoid z rysunku 4 [26].



Rys. 4. Najmniejsze trudne grafy dla *SL*

Fig. 4. The smallest hard graphs for *SL* method

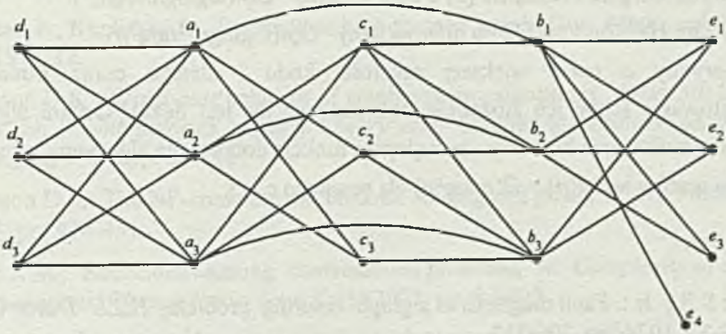
5.3. Metoda SLF

Jeszcze inny sposób porządkowania wierzchołków zaproponował Brélaz [4]. Załóżmy, że wierzchołki grafu G zostały już częściowo pokolorowane. Przez *stopień nasycenia* wierzchołka bezbarwnego v rozumiemy liczbę różnie pokolorowanych wierzchołków sąsiednich do v. W metodzie *SLF* jako v_1 bierzemy wierzchołek o największym stopniu i przydzielamy mu kolor 1, a następnie dla każdego $i=2,...,n$ jako v_i wybieramy wierzchołek o największym stopniu nasycenia i kolorujemy go najniższym możliwym kolorem. Jeżeli jest więcej niż jeden taki wierzchołek, to jako v_i brany jest wierzchołek o największym stopniu $\rho(v_i)$. Tak otrzymane uporządkowanie nazywamy *saturacyjnym LF* lub *SLF* (ang. *Saturation Largest-First*). Algorytm *SLF* może być wykonany w czasie $O(m\log n)$. Można pokazać, że liczba kolorów

$$SLF(G) \leq \max_{i\leq n} \max_{i\leq j} \rho_i(v_j)+1, \tag{11}$$

gdzie $\rho_i(v_j)$ jest stopniem wierzchołka v_j w podgrafie indukowanym przez $v_1,...,v_{i-1},v_j$.

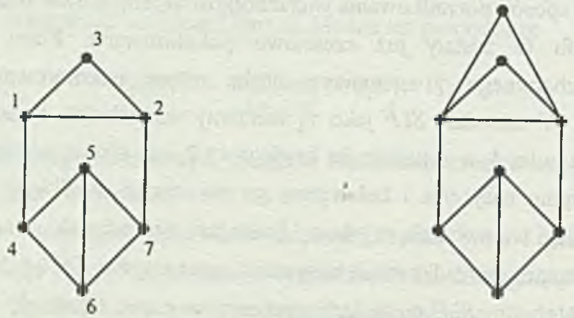
Algorytm *SLF* optymalnie koloruje grafy dwudzielne, koła, cykle i grafy unicykliczne. Jednakże już dla grafów 3-chromatycznych jego rozwiązania mogą być dowolnie odległe od optymalnych. Dobrą ilustracją tego faktu są grafy G_k, $k\geq 3$ o $|V_k|=O(n)$ wierzchołkach, pokazane na rysunku 5. Jeżeli wierzchołki grafu G_k malowane są w kolejności $a_1,b_1,c_1,..,a_k,b_k,c_k,d_1,..,d_k,e_1,..,e_{2k-2}$, to $SLF(G_k)=k+1$. Zatem $SLF(n)=O(n)$ [20].



Rys. 5. Graf G_3 trudny dla *SLF*

Fig. 5. Graph G_3 hard for *SLF* method

Najmniejszy trudny graf i najmniejszy bardzo trudny do kolorowania graf dla metody *SLF* są pokazane na rysunku 6 [27].



Rys. 6. Najmniejsze trudne grafy dla *SLF*

Fig. 6. The smallest hard graphs for *SLF* method

6. Zakończenie

Powyżej opisaliśmy trzy najbardziej znane algorytmy sekwencyjne. Aczkolwiek działają one dość dobrze w przypadku przeciętnym, ich funkcja dobroci jest liniowa, czyli najgorsza możliwa. Przegląd powyższy kończymy wzmianką na temat czterech algorytmów kolorowania mających subliniową funkcję dobroci kolorowania. Są to w kolejności:

1. Algorytm Johnsona [17] o dobroci $A(n) = O(n/\log n)$.
2. Algorytm Wigdersona [30] o dobroci $A(n) = O(n(\log\log n/\log n)^2)$.
3. Algorytm Bergera i Rompela [3] o dobroci $A(n) = O(n(\log\log n/\log n)^3)$.
4. Algorytm Halldórssona [15] o dobroci $A(n) = O(n(\log\log n)^2/\log^3 n)$.

Są to algorytmy o coraz większej zawiłości kodu i dlatego coraz trudniejsze do zaimplementowania (choć ich złożoność obliczeniowa nie jest duża). Ostatni algorytm dał asumpt do sformułowania hipotezy, że najlepsza funkcja dobroci dla algorytmu przybliżonego kolorowania grafów jest $A(n) = \Theta(n/\log^c n)$ dla pewnego $c \geq 3$.

LITERATURA

[1] Akers S.B., Jr.: Fault diagnosis as a graph coloring problem, *IEEE Trans. Comput.*, vol. C-23, 1974, pp. 706-712.

[2] Akers S.B., Jr.: Wyznaczanie ścieżek połączeń, w: *Automatyczne projektowanie maszyn cyfrowych*, *PWN*, Warszawa, 1976, pp. 361-425.

[3] Berger B., Rompel J.: A better performance guarantee for approximate graph coloring, *Algorithmica*, vol. 5, 1990, pp. 459-466.

[4] Brélaz D.: New methods to color the vertices of a graph, *Comm. ACM*, vol. 22, 1979, pp. 251-256.

[5] Brooks R.L.: On colouring the nodes of a network, *Proc. Cambridge Philos. Soc.*, vol. 37, 1941, pp. 194-197.

[6] Burlaga H.: Optymalizacja struktur czaso-przestrzennych metodami kolorowania grafów, Praca doktorska, *WAT*, Warszawa, 1971.

[7] Carvajal R.: Modulus *k*-check digits and the chromatic number problem, *Proc. 6th IFIP Congress*, Stockholm, 1974, pp. 521-526.

[8] Chaitin G.J., Auslander M.A., Chandra A.K., Cocke J., Hopkins M.E., Markstein P.W. Register allocation via coloring, *Comp. Lang.*, vol. 6, 1981, pp. 47-57.

[9] Chung F.R.K., Leighton F.T., Rosenberg A.L.: Embedding graphs in books: A layout problem with application to VLSI design, *SIAM J. Alg. Disc. Meth.*, vol. 8, 1987, pp. 33-58.

[10] Coleman T.F., More J.J.: Estimation of sparse Jacobian matrices and graph coloring problems, *SIAM J. Numer. Anal.*, vol. 20, 1983, pp. 187-209.

[11] Dencker P., Dürre K., Heuft J.: Optimization of parser tables for portable computers, *ACM Trans. Programm. Lang. Sys.*, vol. 6, 1984, pp. 546-572.

[12] Garey M.R., Johnson D.S., So H.C.: An application of graph coloring to printed circuit testing, *IEEE Trans. Circ. Syst.*, vol. CAS-23, 1976, pp. 591-599.

[13] Grimmet G.R., McDiarmid C.J.H.: On colouring random graphs, *Math. Proc. Cambridge Philos. Soc.*, vol. 77, 1975, pp. 313-324.

[14] Gupta U.I., Lee D.T., Leung J.Y.: An optimal solution for the channel-assignment problem, *IEEE Trans. Comput.*, vol. C-28, 1979, pp. 807-810.

[15] Halldórsson M.M.: A still better performance guarantee for approximate graph coloring, *Inf. Process. Lett.*, vol. 45, 1993, pp. 19-23.

[16] Hansen P., Kuplinsky J.: The smallest hard-to-color graph, *Disc. Math.*, vol. 96, 1991, pp. 199-212.

[17] Johnson D.S.: Worst-case behavior of graph coloring algorithms, *Proc. 5th South-East. Conf. on Combinatorics, Graph Theory and Computing, Utilitas Mathematica*, Winnipeg, 1974, pp. 513-527.

[18] Johnson D.S.: The NP-completeness column: An ongoing guide, *J. Algorithms*, vol. 6, 1985, pp. 434-451.

[19] Karp R.M.: Reducibility among combinatorial problems, w: *Complexity of Computer Computations, Plenum Press*, New York, 1972, pp. 85-103.

[20] Kubale M.: Metody kolorowania grafów i ich zastosowanie w wybranych problemach technicznych, *Zesz. Nauk. PG Seria Elektronika*, vol. 65, 1988.

[21] Lawler E.L.: A note on the complexity of the chromatic number problem, *Inform. Process. Lett.*, vol. 5, 1976, pp. 66-67.

[22] Lund C., Yannakakis M.: On the hardness of approximating minimization problems, *Maszynopis*, 1992.

[23] Matula D.W., Marble G., Isaacson D.: Graph coloring algorithms, w: *Graph Theory and Computing, Academic Press*, New York, 1972, pp. 109-122.

[24] McCormick S.T.: Optimal approximation of sparse Hessians and its equivalence to a graph coloring problem, *Math. Programm.*, vol. 26, 1983, pp. 153-171.

[25] Moll U., Schweigert D.: A heuristic algorithm for the frequency assignment problem in mobile phone, *Maszynopis*, 1989.

[26] Pakulski J.: Trudne i bardzo trudne do kolorowania grafy dla metody SL, *Raport techniczny, Katedra Podstaw Informatyki PG*, Gdańsk, 1993.

[27] Pakulski J.: Informacja ustna, Gdańsk, 1994.

[28] Schmitt A.: Minimizing storage space of sparse matrices by graph coloring algorithms, w: *Graphs, Data Structures Algorithms*, Munich, 1979, pp. 157-168.

[29] Welsh D.J., Powell M.B.: An upper bound for the chromatic number of a graph and its application to timetabling problem, *Comput. J.*, vol. 10, 1967, 85-86.

[30] Wigderson A.: Improving the performance quarantee for approximate graph coloring, *J. ACM*, vol. 30, 1983, 729-735.

Recenzent: Prof.dr hab.inż. Andrzej Świerniak

Wpłynęło do Redakcji do 30.04.1994r.

Abstract

Graph coloring problems belong to the hardest combinatorial problems. Recently, it has been shown that designing a relative approximation polynomial-time algorithm for coloring graph vertices is NP-hard. In general, in the graph coloring problem the following tasks are considered: coloring the vertices and coloring the edges of a graph. Although both tasks are NP-complete, the problem of coloring the vertices is harder than that of coloring the edges with respect to effective polynomial-time algorithms. Also, both problems have numerous practical applications.

This article is the first of two review papers on graph coloring problems. We consider herein the problem of coloring the vertices of a graph against a background of practical applications in selected areas of electronics and telecommunication engineering, automation and computer science. In particular, we review the following: bounds on the chromatic number, technological problems in which vertex coloring has been successfully applied and some polynomial approximation algorithms for vertex coloring.