ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1998
Seria: INFORMATYKA z. 35 Nr kol. 1402

Krzysztof DOBOSZ
Politechnika Śląska, Instytut Informatyki

WYKREŚLANIE GRAFÓW CYKLICZNYCH

Streszczenie. W publikacji przedstawiono metodę wykreślania cyklicznych struktur grafowych w przestrzeni dwuwymiarowej. Artykuł obejmuje opis algorytmu wraz z oceną jego efektywności. Poruszono w nim takie problemy, jak znajdowanie fundamentalnego zbioru cykli, wyznaczanie wagi cyklu, zastosowanie figur foremnych do reprezentacji cykli.

DRAWING CYCLIC GRAPHS

Summary. The paper shows the method of drawing cyclic graphs on a two-dimensional surface. The paper comprises both description of the algorithm and evaluation of its efficiency. Problems like finding fundamental set of cycles, calculating cycles' parameters, and application of regular figures to the representation of cycles are discussed.

1. Wstęp

Częstokroć chcąc przeanalizować jakiś problem, staramy się przedstawić go w postaci struktury zbudowanej z węzłów i krawędzi, którą w prosty i przejrzysty sposób można nanieść na kartkę papieru lub zaprezentować na ekranie monitora. Niestety, wielu problemów nie da się opisać tym sposobem, wykorzystując tylko proste struktury drzewiaste. Bardziej elastyczną strukturę stanowi graf [1], w którym nie ma ograniczenia mówiącego, że ciąg krawędzi pomiędzy dowolnymi dwoma węzłami musi być jednoznaczny. Powstaje wówczas problem wykreślenia grafu (w szczególności grafu cyklicznego), w którym nie można określić hierarchii struktury i jednoznacznie wyznaczyć kolejności wyświetlania węzłów i krawędzi. W niniejszym artykule zaproponowano metodę rozwiązania tego problemu.

Wykreślanie grafów polega na konstruowaniu ich graficznej reprezentacji. W zależności od rodzaju struktury grafu oraz metody reprezentacji, wierzchołki i krawędzie grafu mogą zostać rozmieszczone w zadanej przestrzeni na wiele różnych sposobów. W zależności od efektu graficznego, jaki chcemy osiągnąć na ekranie, stosujemy konkretną metodę pozwalającą przedstawić graf zgodnie z przyjętymi założeniami. Zazwyczaj algorytmy wykreślania grafów starają się automatycznie generować ich graficzne reprezentacje spełniające określone warunki. Mogą do nich należeć: minimalizacja przecinających się krawędzi, minimalizacja obszaru zajmowanego przez graf, wykreślanie symetryczne, wykreślanie krawędzi o określonych kształtach itp. Również obszar, w jakim graf ma zostać pokazany, może być nieograniczony lub ograniczony jakąś figurą geometryczną w dwu- lub wielowymiarowej przestrzeni.

Automatyczne wykreślanie struktur grafowych jest niezwykle pomocne w takich dziedzinach, jak: projektowanie sieci, projektowanie połączeń w strukturach VLSI, inżynieria programowania czy tworzenie interfejsów graficznych. Często więc zdarza się, że algorytmy specjalizują się w wykreślaniu struktur grafowych o określonych parametrach. Do parametrów tych można zaliczyć spójność grafu, hierarchiczność, cykliczność, maksymalną liczbę krawędzi przyległych do danego wierzchołka, maksymalnym stosunku liczby krawędzi do liczby wierzchołków w grafie itp. Niestety, specjalizowane algorytmy zazwyczaj zupełnie nie nadają się do wykreślania struktur grafowych, które założeń takich nie spełniają.

Badania nad wykreślaniem struktur grafowych zintensyfikowały się w ostatnich latach. Ostatnie oficjalne zestawienie publikacji na temat wykreślania grafów zawiera ponad 300 pozycji [2]. Rozwijane zagadnienia są związane ze wszelkimi typami grafów oraz wieloma możliwymi sposobami ich wykreślania. Zazwyczaj nacisk kładzie się na rozwiązania charakteryzujące się generowaniem minimalnej liczby przecinających się krawędzi [3], zajętością przez graf minimalnego obszaru [4], wyświetlaniem krawędzi w postaci lini prostych [5,6] bądź łamanych [7] lub też silnym naciskiem na symetrię [8,9]. Niniejszy artykuł przedstawia nową metodę starająca się pogodzić niektóre wymagania stawiane wykreślanym grafom. Jest ona rezultatem badań autora nad wykreślaniem grafów cyklicznych. Informacje na temat wykorzystywanej terminologii związanej z teorią grafów można znaleźć w [1,10,11].

2. Charakterystyka dziedziny wykreślanych grafów

W trakcie opracowywania metody przyjęto pewne założenia co do struktury wykreślanego grafu, pozwalające na sprecyzowanie sposobu jego konstruowania. Opisywany algorytm zaimplementowano opierając się na macierzy połączeń wierzchołków grafu. Dlatego też

niektóre z założeń w istotny sposób wpływają na zawartość tej macierzy, a co za tym idzie, na złożoność przeszukujących ją algorytmów.

Tak więc każdy graf wykreślany za pomocą omawianej metody powinien mieć następujące cechy:

- spójność - pomiędzy każdą parą wierzchołków jest przynajmniej jedna łącząca je ścieżka. Unikamy dzięki temu tworzenia na ekranie dwóch lub więcej niezależnych obiektów graficznych,
- nieskierowalność - krawędzie w garfie nie są zorientowane, a odpowiadające im pary wierzchołków są nieuporządkowane,
- brak krawędzi równoległych - pomiędzy każdą parą dwóch wierzchołków istnieje co najwyżej jedna krawędź. Oznacza to, że macierz połączeń wierzchołków grafu wypełniona jest wyłącznie wartościami 0 i 1. Istnienie krawędzi równoległych nie wnosi do rozpatrywanego zadania żadnych nowych istotnych problemów,
- brak zapętleń - nie ma krawędzi łączących dowolny wierzchołek grafu z nim samym. Oznacza to, że przekątna macierzy połączeń wierzchołków grafu wypełniona jest zerami,
- cykliczność - w grafie powinien istnieć co najmniej jeden cykl, czyli zamknięty obwód składający się z krawędzi grafu.

3. Analiza cykli w grafie

Cyklem lub inaczej obwodem nazywamy ścieżkę w grafie, która rozpoczyna się i kończy tym samym wierzchołkiem [10,11]. Wykreślanie poszczególnych cykli jest istotą działania omawianego w niniejszym artykule algorytmu. Dlatego też pierwszym etapem działania algorytmu jest znalezienie fundamentalnego zbioru cykli [10] i posortowanie jego składowych zgodnie z ustaloną zależnością.

3.1. Wyszukiwanie cykli w macierzy połączeń wierzchołków

Znajdowanie zbioru cykli fundamentalnych jest ściśle związane z problemem generowania drzewa rozpinającego w grafie. Drzewo takie jest podgrafem grafu analizowanego, zawiera wszystkie jego wierzchołki. Do tego zadania można zastosować zarówno metodę przeszukiwania grafu w głąb, jak i wszerz. Aby otrzymać cykl podstawowy, wystarczy do powstałego drzewa rozpinającego dodać dowolną cięciwę. Podczas implementacji sposób ten może nastręczać pewne trudności. Dlatego alternatywną metodą znajdowania fundamentalnego zbioru cykli jest metoda polegająca na znalezieniu zbioru

wszystkich cykli i usunięciu z niego cykli nadmiarowych. Metoda ta jest bardziej złożona czasowo, lecz łatwiejsza do zaimplementowania.

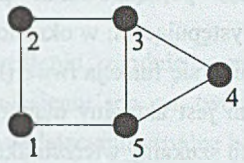
Problem w znajdowaniu cykli polega na tym, że nie można określić, w którym miejscu cykl się zaczyna, a więc nie ma wierzchołka początkowego ani końcowego. Najłatwiejszym sposobem jest przyjęcie, że wierzchołek o najmniejszym numerze będzie wierzchołkiem początkowym cyklu. Opierając się na tym założeniu, już na wstępie możemy eliminować wygenerowane cykle, w których wierzchołek o najmniejszym numerze nie jest wierzchołkiem początkowym. W celu dalszego uproszczenia algorytmu przeszukującego macierz sąsiedztwa wierzchołków przekształcamy wspomnianą tablicę tak, aby dla dowolnych i, j zachodziło: $t[i,j] * t[j,i] = 0$.

Zakładając, że macierz wypełniona jest jak dla grafów nieskierowanych (symetrycznie względem przekątnej $i = j$), uzyskujemy w ten sposób to, że wszystkie elementy macierzy położone poniżej tej przekątnej wypełniamy wartością 0. Następnie stosując przeszukiwanie w głąb lub wszerz po polach wypełnionych wartością 1, znajdujemy kolejno wszystkie cykle. Kierunki poszukiwań pokazano na rys.1. Pamiętając o tym, że wierzchołkiem początkowym cyklu jest wierzchołek o najmniejszym numerze, podczas przeszukiwania tablicy w celu znalezienia cyklu zaczynającego się od wierzchołka i nie musimy odwiedzać rzędów ani kolumn o numerach mniejszych od i. Powrót do rzędu lub kolumny o numerze i oznacza zamknięcie ścieżki i znalezienie cyklu.

	1	2	3	4	5	6	7	8
1	0	0	1	0	1	0	1	0
2	0	0	0	0	0	0	0	
3	0	0	0	0	1	0		
4	0	0	0	0	0			
5	0	0	0	0				
6	0	0	0					
7	0	0						
8								

Rys. 1. Przeszukiwanie w tablicy sąsiedztwa wierzchołków
Fig. 1. Searching in the array of vertices neighbourhood

Teraz ze zbioru cykli zostają usunięte nadmiarowe, tak aby otrzymać podstawowy zbiór cykli. Przez cykl nadmiarowy rozumiemy w tym przypadku cykl, którego wszystkie wierzchołki należą cykli podstawowych o mniejszej od niego liczbie wierzchołków (długości).



Rys. 2. Przykładowy graf cykliczny
Fig. 2. Sample cyclic graph

Na rys. 2 przedstawiono przykładowy graf. Zawiera on następujące cykle:

C1 = (1, 2, 3, 5, 1), C2 = (3, 4, 5, 3) i C3 = (1, 2, 3, 4, 5, 1). Cykl C1 ma długość 4 i nie jest nadmiarowy, ponieważ tylko dwa z czterech jego wierzchołków należą do cyklu C2 o długości 3. Cykl C2 nie jest nadmiarowy, ponieważ charakteryzuje się najmniejszą długością w zbiorze wszystkich cykli przykładowego grafu. Oznacza to, że jego wierzchołki nie mogą należeć do cykli o mniejszej od niego długości. Cykl C3 jest nadmiarowy, ponieważ każdy z jego wierzchołków należy do C1 lub do C2, których długości są mniejsze od 5.

W wyniku procesu minimalizacji cykli, ze zbioru cykli grafu G zostają usunięte wszystkie cykle nadmiarowe. Pozostaje nam zbiór cykli fundamentalnych.

3.2. Sortowanie cykli

Cykle zostają umieszczone na liście i tam posortowane. Pierwszy cykl zostaje wyznaczony na podstawie wagi. Wagę cyklu możemy obliczać na różne sposoby. Od tego sposobu zależeć będzie wybór pierwszego cyklu do wykreślenia, a to pociągnie za sobą kolejność rysowania pozostałych cykli, a więc również końcowy efekt, który otrzymamy na ekranie. Zaproponowano tu metodę biorącą pod uwagę liczbę krawędzi przylegających do wierzchołków cyklu.

Algorytm obliczania wagi cyklu wygląda następująco:

suma := 0;
for i := 1 to liczba_wierzchołków_grafu do
  suma := suma + liczba_krawędzi_przyległych_do_wierzchołka(i) - 2;
waga := suma / liczba_wierzchołków;

Waga cyklu policzona tym sposobem oznacza stosunek krawędzi przylegających do wierzchołków cyklu, ale do tego cyklu nie należących, do liczby wierzchołków cyklu.

Cykl o największej wadze zostanie wykreślony jako pierwszy. Po jego wyznaczeniu przystępujemy do porządkowania pozostałych cykli. W pierwszej kolejności zakładamy, że aby cykl mógł zostać umieszczony w obszarze wyświetlania, co najmniej jeden z jego wierzchołków musi się już tam znajdować. Wierzchołek taki jest wtedy punktem zaczepienia

dorysowywanego cyklu. Podczas porządkowania preferowane są cykle zawierające największą ilość wierzchołków występujących w określonej liczbie cykli ją poprzedzających. Obliczeniem tego parametru zajmuje się funkcja *lwwc* (liczba wykorzystanych wierzchołków cyklu). Pierwszym jej parametrem jest aktualny numer badanego cyklu, drugim zaś liczba początkowych cykli, wśród których szukamy wierzchołków cyklu badanego.

Jeśli wartości funkcji *lwwc* dla dwóch cykli są identyczne, to przewagę ma cykl o większej wadze. Gdy również wagi tych cykli są identyczne, to wtedy preferowany jest cykl o większej długości. Jest to związane z tym, że cykl o dużej długości tym trudniej narysować w postaci wieloboku foremnego, im więcej jego wierzchołków należących do innych cykli będzie już porozsiewanych po ekranie. Dlatego należy dążyć do tego, aby duże cykle rysować stosunkowo wcześnie.

Algorytm porządkowania wygląda następująco:

```
// znajdujemy pierwszy cykl
for i := liczba_cykli downto 2 do
    if ( waga_cyklu( i ) > waga_cyklu( i - 1 ) )
        or ( ( waga_cyklu( i ) = waga_cyklu( i - 1 ) )
        and ( długość_cyklu( i ) > długość_cyklu( i - 1 ) ) ) then
    zamiana_cykli_miejscami;

// porządkujemy pozostałe cykle
for i := 2 to liczba_cykli -1 do
    for j := liczba_cykli downto i + 1 do
    begin
    if ( lwwc(j, i ) > lwwc(j - 1, i ) then
        zamiana_cykli_miejscami;
    if ( lwwc(j, i ) = lwwc(j - 1, i ) then
        if ( waga_cyklu( i ) > waga_cyklu( i - 1 ) ) then
            or ( ( waga_cyklu( i ) = waga_cyklu( i - 1 ) )
            and ( długość_cyklu( i ) > długość_cyklu( i - 1 ) ) ) then
        zamiana_cykli_miejscami;
    end;
```

4. Wykreślanie

Wykreślenie grafu polega na określeniu, zgodnie z przyjętymi założeniami, położenia każdego z jego wierzchołków i naniesieniu ich na tworzony rysunek oraz wykreśleniu połączeń pomiędzy wierzchołkami posiadającymi wspólne krawędzie.

Do najważniejszych problemów, które trzeba rozwiązać podczas wykreślania grafu, należy zaliczyć:

- określenie kolejności wykreślania wierzchołków,
- określenie kolejności wykreślania krawędzi (dotyczy metod, w których krawędzie nie są reprezentowane przez odcinek prostoliniowy),
- wyśrodkowanie obrazu struktury w dostępnym obszarze wykreślania.

4.1. Założenia

Zaproponowana poniżej nowa metoda wykreślania grafów ma służyć do konstruowania ich graficznej reprezentacji w przestrzeni dwuwymiarowej. Nie założono żadnych ograniczeń tej przestrzeni. Metoda ta ma na celu wykreślenie grafu w zwartej postaci, przy jednoczesnym unikaniu niepotrzebnego krzyżowania się krawędzi i zbyt bliskiego położenia przyległych wierzchołków. W celu spełnienia powyższych wymagań zostały przyjęte następujące założenia:

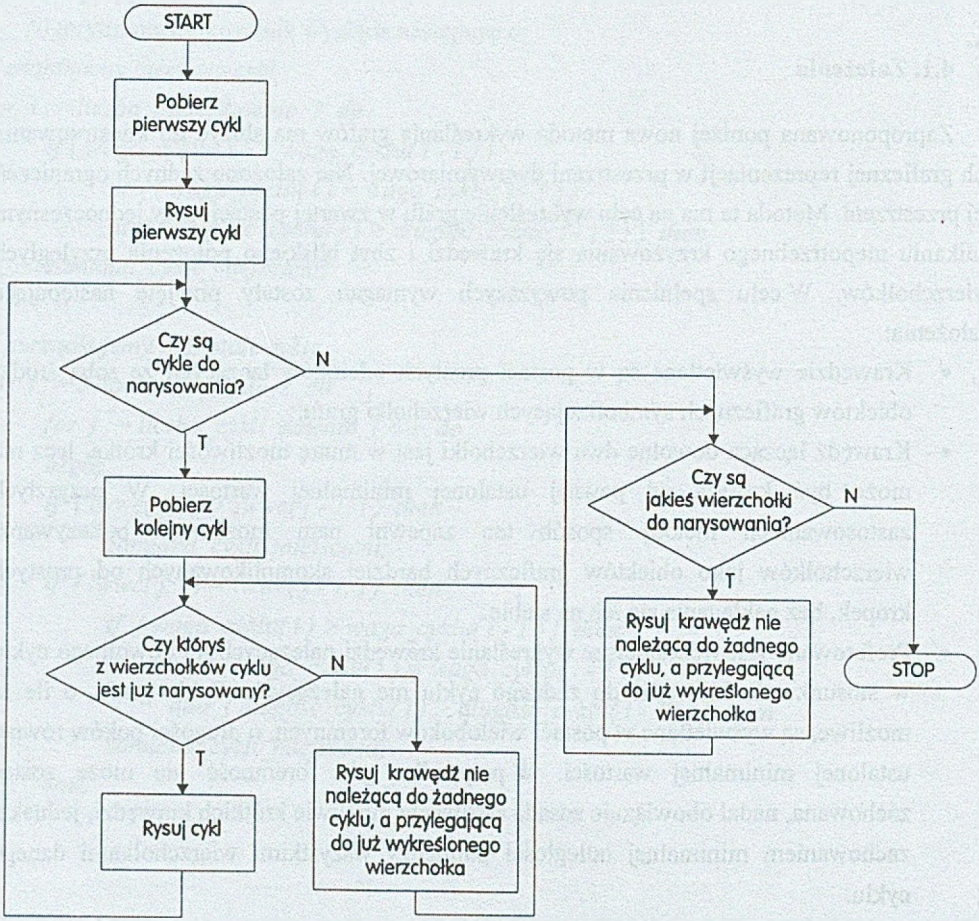
- Krawędzie wyświetlane są w postaci prostych odcinków łączących ze sobą środki obiektów graficznych symbolizujących wierzchołki grafu.
- Krawędź łącząca dowolne dwa wierzchołki jest w miarę możliwości krótka, lecz nie może być krótsza od pewnej ustalonej minimalnej wartości. W przyszłych zastosowaniach metody sposób ten zapewni nam możliwość pokazywania wierzchołków jako obiektów graficznych bardziej skomplikowanych od prostych kropek, bez nakładania się ich na siebie.
- Preferowane jest wcześniejsze wykreślanie krawędzi należących do dowolnego cyklu w stosunku do krawędzi do żadnego cyklu nie należących. Cykle grafu, o ile to możliwe, są wyświetlane w postaci wieloboków foremnych, o długości boków równej ustalonej minimalnej wartości. W przypadku gdy foremność nie może zostać zachowana, nadal obowiązuje zasada rysowania możliwie krótkich krawędzi, jednak z zachowaniem minimalnej odległości pomiędzy wszystkimi wierzchołkami danego cyklu.
- Pierwszy cykl z uporządkowanej wcześniej listy cykli grafu jest rysowany w postaci figury foremnej. Kolejne elementy grafu dorysowywane są na zewnątrz tego cyklu, w miarę możliwości symetrycznie do prostych przechodzących przez jego środek.

- Raz narysowane wierzchołki i krawędzie nie są przesuwane i pozostają na swoim miejscu aż do końca wykreślania całego grafu.

4.2. Algorytm

Proces wykreślania grafu składa się z kombinacji dwóch procesów składowych:

- rysowania wszystkich nie wyświetlonych jeszcze krawędzi należących do danego cyklu;
- rysowania krawędzi nie należących do żadnego cyklu, ale przyległych do wierzchołków już wyświetlonych;

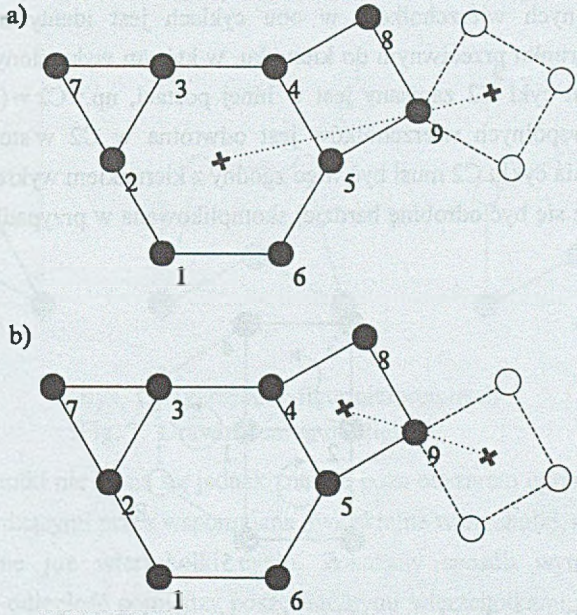


Rys. 3. Algorytm wykreślania grafu
Fig. 3. Algorithm for graph drawing

Algorytm działa zgodnie z założeniem, że wykreślany graf jest spójny. Pierwszy cykl zostaje wykreślony symetrycznie wokół środka dostępnego obszaru, natomiast kolejne cykle są rysowane wtedy, gdy co najmniej jeden należący do nich wierzchołek jest już wykreślony. Jeśli brak takich wierzchołków, dorysowywana jest dowolna krawędź (wraz z jednym wierzchołkiem) nie należąca do żadnego z cykli, a przylegająca do jednego z już wykreślonych wierzchołków. Następnie znów podejmowana jest próba wykreślenia, czekającego w kolejce do wyrysowania, cyklu.

4.3. Cykle w postaci wielokątów foremnych

Cykl może zostać wyrysowany w postaci wieloboku foremnego wtedy, gdy dokładnie jeden z jego wierzchołków znajduje się już na ekranie. Na podstawie liczby wierzchołków danego cyklu oraz położenia środka symetrii pierwszego wyświetlonego cyklu grafu (zwanego dalej środkiem grafu) wyznaczamy środek wieloboku jak na rys.4a. Środek wieloboku można też obliczyć na podstawie położenia środka przyległego cyklu, rys.4b. Wykreślany graf jest wtedy bardziej zwarty, lecz jest większa szansa na przecięcie się krawędzi przy dorysowywaniu kolejnych cykli grafu.

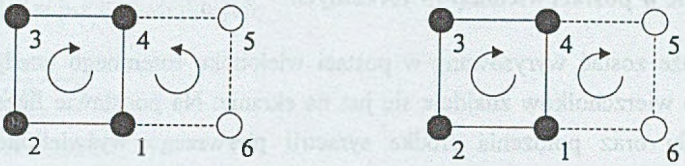


Rys. 4. Znajdowanie środka cyklu, a) z wykorzystaniem środka grafu, b) z wykorzystaniem środka jego sąsiedniego cyklu

Fig. 4. Finding the center of a cycle, a) using the center of the graph, b) using the center of its neighbour cycle

W powyższych przypadkach kierunek wykreślania wierzchołków cyklu jest bez znaczenia i przyjmowano zazwyczaj zgodny z ruchem wskazówek zegara.

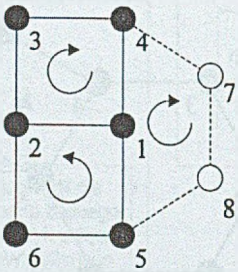
Często zachodzi możliwość wykreślenia cyklu w postaci foremnego wieloboku, wtedy gdy dwa jego wierzchołki umieszczone już są na ekranie. Wielobok wykreślany jest jako foremny wtedy, gdy długość krawędzi pomiędzy tymi dwoma wierzchołkami jest równa przyjętej minimalnej odległości (patrz punkt 4.2). Pozostaje teraz jeszcze istotna dla procedury wykreślającej kwestia rozstrzygnięcia kierunku rysowania cyklu.



Rys. 5. Znajdowanie kierunku wykreślania cyklu - przykład 1
Fig. 5. Finding direction of drawing cycle - example 1

Rozpatrzmy sytuację przedstawioną na rys.5. Mamy dane dwa cykle: C1 = (1, 2, 3, 4, 1) oraz C2 = (1, 6, 5, 4, 1), przy czym C1 jest już wykreślony. W takim przypadku, gdy sekwencja wspólnych wierzchołków w obu cyklach jest identyczna, cykl C2 zostaje wykreślony w kierunku przeciwnym do kierunku, w którym wykreślony został cykl C1. Gdy natomiast ten sam cykl C2 zapisany jest w innej postaci, np.: C2 = (1, 4, 5, 6, 1), wtedy sekwencja obu wspólnych wierzchołków jest odwrotna w C2 w stosunku do cyklu C1. Kierunek rysowania cyklu C2 musi być więc zgodny z kierunkiem wykreślania cyklu C1.

Sytuacja zdaje się być odrobinę bardziej skomplikowana w przypadku sytuacji pokazanej na rys.6.

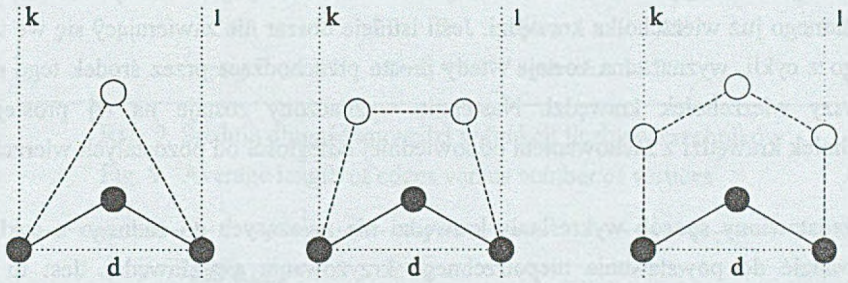


Rys. 6. Znajdowanie kierunku wykreślania cyklu - przykład 2
Fig. 6. Finding direction of drawing cycle - example 2

Tutaj bowiem cykl C3 = (1, 4, 7, 8, 5, 1), który mamy zamiar wykreślić, sąsiaduje z dwoma cyklami C1 = (1, 2, 3, 4, 1) oraz C2 = (1, 2, 6, 5, 1). W takiej sytuacji rozpatrujemy sąsiedztwo tylko z jednym, dowolnym cyklem (pod warunkiem, że cykl, który chcemy dorysować, ma z nim co najmniej dwa wspólne wierzchołki). Następnie określamy sekwencję wspólnych wierzchołków i dalej postępujemy już tak, jak w przypadku rozpatrzonym wcześniej.

4.4. Cykle w postaci wielokątów nieforemnych

Cykle wykreślane są w postaci wielokątów nieforemnych wtedy, gdy dwa lub więcej wierzchołków bieżącego cyklu jest już na ekranie, a ich położenie uniemożliwia wykreślenie opartej na nich foremnej figury o długości boków równych ustalonej minimalnej wartości. W takim przypadku, kosztem zniekształcenia foremnej figury, staramy się wyświetlić cykl jako figurę o minimalnej długości boków w kierunku przeciwnym do miejsca położenia cykli przylegających. Sposób wyświetlenia przedstawiono na rys.7. Najpierw wyznaczamy odcinek *d* łączący dwa najbardziej skrajne z już wyświetlonych wierzchołków cyklu. Następnie wykreślamy dwie proste *k* i *l* przechodzące przez te wierzchołki, prostopadłe do odcinka *d*. Położenie dorysowywanych wierzchołków jest uzależnione od ich liczby.

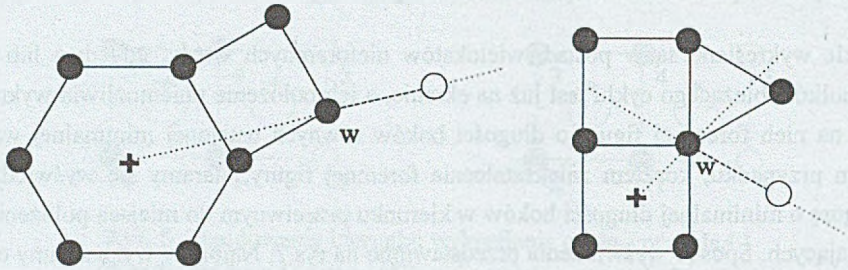


Rys. 7. Wykreślanie figur nieforemnych
Fig. 7. Drawing unregular figures

Rysowane wierzchołki nie mogą się jednak znaleźć poza obszarem ograniczonym dwoma prostymi *k* i *l*, przechodzącymi przez wspomniane dwa skrajne wierzchołki, oraz krawędziami łączącymi wyświetlone już wierzchołki cyklu. Pokazany sposób wynika z założenia mówiącego o tym, że odległość pomiędzy poszczególnymi wierzchołkami ma być możliwie jak najkrótsza, jednak nie mniejsza od ustalonej minimalnej wartości.

4.5. Krawędzie nie należące do żadnych cykli

W grafach cyklicznych mogą występować również krawędzie pomiędzy wierzchołkami nie należące do żadnego z cykli (sytuacja ta nie występuje w grafach co najmniej podwójnie spójnych). Krawędź taka jest wykreślana na prostej przechodzącej przez środek pierwszego cyklu i wierzchołek, do którego jest przyległa. Drugi wierzchołek rysowany jest na tej prostej w takim miejscu, aby jego odległość od dowolnego z wykreślonych już wierzchołków grafu była możliwie mała, lecz nie mniejsza niż określone minimum.



Rys. 8. Wykreślanie krawędzi
Fig. 8. Drawing edges

Na rys.8 przedstawiono również sytuację, gdy wspomniana prosta przecina jedną z krawędzi grafu leżących na kierunku wykreślania. Badany jest wtedy obszar dookoła wykreślonego już wierzchołka krawędzi. Jeśli istnieje obszar nie zawierający się we wnętrzu żadnego z cykli, wyznaczona zostaje wtedy prosta przechodząca przez środek tego obszaru i pierwszy wierzchołek krawędzi. Następnie wyznaczony zostaje na tej prostej drugi wierzchołek krawędzi z zachowaniem odpowiedniej odległości od pozostałych wierzchołków grafu.

Przedstawiony sposób wykreślania krawędzi nie należących do żadnego z cykli może doprowadzić do powstawania niepotrzebnego krzyżowania się krawędzi. Jest to jednak wynikiem założenia, że zachowanie minimalnej odległości pomiędzy wierzchołkami ma większy priorytet niż przestrzeganie planarności wykreślanego grafu.

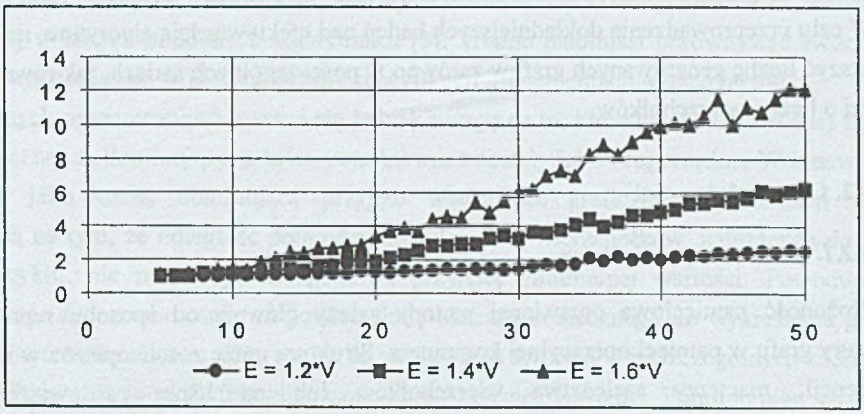
5. Efektywność metody

5.1. Charakterystyka otrzymanych wyników

W celu zbadania efektywności metody przeprowadzono pomiary na losowo wygenerowanych grafach spełniających kryteria z rozdziału 2. Dla każdego V (gdzie $V = 5, 6, \ldots 50$), wygenerowano 100 różnych V-wierzchołkowych grafów i policzono badaną

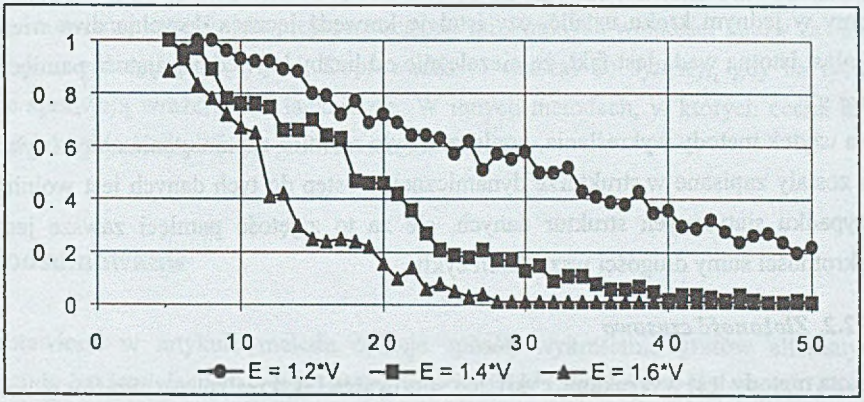
zależność. Na wykresach przedstawiono średnią arytmetyczną wszystkich wyników dla danej wielkości *V*.

Wykresy tego typu najczęściej wykorzystywane są do porównywania różnych metod wykreślania grafów [12]. W niniejszym artykule zademonstrowano dwie zależności: na rys.9 pokazano zależność średniej długości krawędzi grafu w zależności od liczby jego wierzchołków, natomiast na rys.10 stosunek liczby cykli przedstawionych w postaci figur foremnych do liczby wszystkich podstawowych cykli grafu w funkcji liczby jego wierzchołków. Na każdym z wykresów rozpatrzono trzy rodzaje grafów charakteryzujące się różną ilością liczby krawędzi *E* w stosunku do liczby wierzchołków *V*. W porównaniach, gdzie występuje długość krawędzi, jako jednostkę przyjęto minimalną, nieprzekraczalną odległość pomiędzy dwoma przyległymi wierzchołkami.



Rys. 9. Średnia długość krawędzi w funkcji liczby wierzchołków

Fig. 9. Average length of edges versus number of vertices



Rys. 10. Średni odsetek figur foremnych w funkcji liczby wierzchołków

Fig. 10. Average percent of regular figures versus number of vertices

Jak widać na załączonych wykresach, algorytm daje dość dobre rezultaty dla grafów o liczbie krawędzi nie przekraczającej zbytnio liczby wierzchołków. Znaczny procent cykli przedstawionych w postaci foremnych figur geometrycznych powoduje, że otrzymany rysunek jest estetyczny i łatwy w odbiorze dla użytkownika. Bliskie położenie przyległych wierzchołków pozwala na szybkie wzrokowe prześledzenie wzajemnych powiązań pomiędzy wierzchołkami.

Niestety, w miarę zwiększania liczby krawędzi w grafie, pogarszają się efekty działania algorytmu. Dla przypadku gdy stosunek liczby krawędzi do liczby wierzchołków w grafie wynosi 1.6, ilość cykli wyświetlonych w postaci figur foremnych w stosunku do liczby wszystkich cykli podstawowych grafu spada poniżej wartości 0.5 już dla grafu o 12 wierzchołkach. Wraz ze zwiększaniem liczby wierzchołków takich grafów, znacznie rośnie też średnia długość krawędzi, powodując zmniejszenie czytelności wykreślonej figury.

W celu przeprowadzenia dokładniejszych badań nad efektywnością algorytmu, należałoby zwiększyć liczbę generowanych grafów zarówno w poszczególnych seriach, jak również jeśli chodzi o liczbę wierzchołków.

5.2. Ocena złożoności

5.2.1. Złożoność pamięciowa

Złożoność pamięciowa omawianej metody zależy głównie od sposobu reprezentacji struktury grafu w pamięci operacyjnej komputera. Struktura może zostać opisana w macierzy incydencji, macierzy sąsiedztwa wierzchołków lub na liście par wierzchołków odpowiadających poszczególnym krawędziom [10]. Do implementacji wykorzystano macierz sąsiedztwa wierzchołków. Jest to macierz o wymiarach $n*n$, gdzie n oznacza liczbę wierzchołków grafu. Główną zaletą takiej reprezentacji pamięciowej grafu jest fakt, że możemy w jednym kroku ustalić, czy istnieje krawędź łącząca dowolne dwa wierzchołki. Natomiast istotną wadą jest fakt, że niezależnie od liczby krawędzi, zajętość pamięci wynosi zawsze n^2.

Na użytek metody wykreślania, konieczne było również zapamiętanie zbioru cykli. Cykle grafu zostały zapisane w strukturze dynamicznej. Dostęp do tych danych jest wolniejszy niż w przypadku statycznych struktur danych, ale za to zajętość pamięci zawsze jest równa wielokrotności sumy długości wszystkich cykli.

5.2.2. Złożoność czasowa

Istotą metody jest wyszukanie cykli, ich sortowanie i wykreślenie w określonej kolejności. W operacji wyszukiwania cykli w tablicy sąsiedztwa wierzchołków zastosowano metodę przeszukiwania w głąb, którego złożoność czasowa jest rzędu $O(m+n)$, gdzie n oznacza liczbę

wierzchołków, a m - liczbę krawędzi grafu [10]. Biorąc pod uwagę założenia co do dziedziny wyświetlanych grafów (rozdział 2), można przyjąć, że również liczba krawędzi m będzie rzędu $O(n)$.

Wykreślanie grafu polega na sekwencyjnym pobieraniu węzłów i krawędzi, a następnie umieszczaniu ich na tworzonym rysunku. Tutaj również złożoność czasowa tych operacji wynosi $O(m+n)$.

Pozostałe operacje w metodzie wykonywane są na pojedynczych cyklach i złożoności tych operacji nie można w sposób jednoznaczny uzależnić od liczby wierzchołków grafu n. Ogólnie można więc przyjąć, że algorytm posiada złożoność $O(m+n)$.

5.2.3. Porównanie z innymi metodami

Złożoność czasowa omawianej metody $O(m+n)$ odpowiada złożoności czasowej spotykanej w innych podobnych algorytmach [3]. Trudno natomiast porównywać złożoność pamięciową, ponieważ w literaturze brak specyfikacji implementacji innych metod.

W innych opracowaniach zwraca się jednak uwagę na to, które cechy wykreślonej figury geometrycznej są dominujące, a które potraktowane zostały jako drugorzędne. W omawianej metodzie jako cechę dominującą przyjęto właściwość graficznej reprezentacji grafu polegającą na tym, że odległość pomiędzy dowolną parą wierzchołków, należących do tego samego cyklu, nie może być mniejsza od przyjętej minimalnej wartości. Powoduje to zmniejszenie zapotrzebowania na powierzchnię obszaru koniecznego do wykreślenia grafu. Postawiono też stosunkowo duży nacisk na zachowanie symetrii w poszczególnych częściach rysunku. Pozwoliło to na zminimalizowanie liczby przecięć krawędzi. Pomimo tego zdarzają się jednak przypadki grafów planarnych [11], których rysunek wykonany za pomocą omawianej metody zawiera pewną ilość takich właśnie przecięć. Wynika to z założenia (rozdział 4) mówiącego, że krawędzie reprezentowane będą przez odcinki prostoliniowe.

Wadą omawianej metody jest nieprzestrzeganie minimalnych wielkości kątów zawartych pomiędzy krawędziami przyległymi. Doprowadza to czasem do sytuacji, gdy na rysunku krawędzie sprawiają wrażenie nakładania się. W innych metodach, w których cecha ta jest traktowana jako dominująca [13], problem ten nie istnieje.

6. Podsumowanie

Przedstawiona w artykule metoda opisuje sposób wykreślania grafów alternatywny w stosunku do już istniejących. Jest on dobry do przedstawiania grafów cyklicznych o niezbyt dużej liczbie krawędzi. Starano się tu pogodzić takie cechy wykreślanej figury geometrycznej, jak: przejrzystość, minimalizacja przecinających się krawędzi, minimalizacja obszaru

zajmowanego przez graf czy wykreślanie symetryczne. Starano też się, aby jak największą liczbę cykli przedstawić w postaci figur foremnych. Spełnienie tego warunku pozwala na zachowanie przejrzystego obrazu struktury grafowej.

W omawianej metodzie przyjęto, że krawędzie wykreślanego grafu są liniami prostymi. W czasie ulepszania algorytmu można by pokusić się o zastosowanie linii łamanych bądź łukowych. Zaletą takiego rozwiązania będzie zmniejszenie liczby przecinających się krawędzi, a więc zwiększenie czytelności grafu. Odbywa się to jednak kosztem znacznego zwiększenia złożoności obliczeń podczas wykreślania krawędzi.

W dziedzinie wykreślania grafów cyklicznych w dalszym ciągu istnieją nie rozwiązane jeszcze problemy [2,14]. Należy do nich przede wszystkim dynamiczne rysowanie grafów. Ma ono zastosowanie w systemach, które pozwalają użytkownikowi na interaktywne modyfikowanie grafu przez dodawanie i usuwanie wierzchołków i krawędzi. W przypadku gdy dynamicznie zmienia się struktura grafu, jego obraz na ekranie również powinien podlegać płynnym zmianom. Zmiany te jednak nie mogą doprowadzić do powstania zawiłej, zniekształconej i nieczytelnej dla odbiorcy figury geometrycznej. Ściśle związany jest z tym m. in. problem dynamicznego sprawdzania planarności grafu oraz dynamicznej kompresji części jego struktury.

LITERATURA

1. Drozdek A., Simon D.L.: Struktury danych w języku C. WNT, Warszawa 1996.
2. Di Battista G., Eades P., Tamassia R., Tollis I.G.: Algorithms for Drawing Graphs: an annotated bibliography. Comput. Geom. Theory Appl. 1994, t. 4, s.235-182.
3. Kant G. Algorithms for Drawing Planar Graphs. PhD Thesis, Department of Computer Sciences, University Utrecht, Netherlands, 1993.
4. Di Battista G., Tamassia R., Tollis I.G.: Area requirement and symmetry display of planar upward drawings. Discrete Computational Geometry 1992, t. 7, s. 381-401.
5. Lipton R.J., North S.C., Sandberg J.S.: A Method for Drawing Graphs. Proc. ACM Symp. on Computational Geometry 1985, s. 153-166.
6. Eades P.: A Heuristic for Graph Drawing. Congressus Numerantium 1984 t. 42, s. 149-160.
7. Trickey H.: Drag: A Graph Drawing System. Proc. Internat. Conf. on Electronic Publishing, Cambridge University Press 1988, s. 171-181.
8. Manning J.B: Geometric Symmetry in Graphs. PhD Thesis, Department of Computer Sciences, Purdue University, 1990.

9. Eades P., Lin X.: Spring Algorithms and Symmetry. Technical Report, Department of Computer Science, The University of Newcastle, Australia, 1996.
10. Lipski W.: Kombinatoryka dla programistów. WNT, Warszawa 1982.
11. Reingold E. M., Nievergelt J., Deo N.: Algorytmy kombinatoryczne. PWN, Warszawa 1985.
12. Di Battista G., Garg A., Liotta G., Tamassia R., Tassinari E., Vargiu F.: An Experimental Comparison of Three Graph Drawing Algorithms. In Proc. 11Th Annu. ACM Symposium Computational Geometry 1995, s. 306-315.
13. Tamassia R.: Planar Orthogonal Drawings of Graphs. Proc. IEEE Int. Symp. on Circuits and Systems, 1990.
14. Garg A., Tamassia R.: Advances in Graph Drawing. Research Report, Department of Comp. Sc., Brown University, Providence, 1993.

Recenzent: Doc. dr hab. inż. Adam Mrózek

Wpłynęło do Redakcji 7 maja 1997 r.

Abstract

The paper shows the method of drawing cyclic graphs on a two-dimensional surface. The second chapter presents the criteria which have to be fulfilled by graphs drawn by the discussed method. Then in chapter 3, the analysis of cycles is provided. There is the way of obtaining a fundamental cycles' set and of arranging them in order of priority being drawn. The method of graphical construction of graphs is described in chapter 4. At first, the conditions are listed which should be fulfilled by the drawn geometrical figure, and then more interesting connections between its part elements are discussed. That chapter is ilustrated with figures 4 - 7. The last part of the paper concentrates on discussion of efficiency of the decribed method. The figures 8 and 9 showing the parameters of a drawn geometrical figures for different kinds of graphs is provided. Finally, still open problems of drawing cyclic graphs are pointed out.