

Faktoryzacja iloczynu kartezjańskiego grafów

Andrzej Kawula
Promotor: dr Monika Pilśniak

30.06.2018

Spis treści

1	Wprowadzenie	2
2	Wstęp	2
2.1	Informacje wstępne	2
2.2	Kolorowanie produktu iloczynu kartezjańskiego grafów	2
2.3	Lemat o kwadracie	3
2.4	Lemat o izomorfizmie	4
2.5	Lemat o udoskonaleniu faktoryzacji iloczynu kartezjańskiego	4
3	Faktoryzacja z dodatkowymi informacjami	5
3.1	Algorytm kolorowania krawędzi	5
3.2	Nadawanie współrzędnych wierzchołkom	7
4	Etykietowanie produktu kartezjańskiego	7
4.1	Etykietowanie produktu w czasie liniowym	10
4.2	Sprawdzanie spójności kolorowanie właściwego produktu iloczynu kartezjańskiego	10
5	Opis algorytmu faktoryzacji	10
6	Implementacja Algorytmu	10
6.1	Wprowadzenie	10
6.2	Dane wejściowe	10
6.3	Opis pakietów i ważniejszych klas	10
6.4	Opis algorytm faktoryzacji	10

1 Wprowadzenie

2 Wstęp

2.1 Informacje wstępne

Produktem kartezjańskim $G_1 \square G_2$ grafów $G_1 = (V_1, E_1)$ i $G_2 = (V_2, E_2)$ nazywamy graf $G = (V, E)$, którego zbiorem wierzchołków jest iloczyn kartezjański wierzchołków grafów G_1 i G_2 ($V = V_1 \times V_2$), natomiast wierzchołki (x_1, y_1) oraz (x_2, y_2) są połączone w grafie G jeżeli $x_1 = x_2$ oraz $y_1 y_2 \in E_2$ lub $x_1 x_2 \in E_1$ oraz $y_1 = y_2$.

Iloczyn kartezjański grafów jest działaniem łącznym, przemennym, z dokładnością do izomorfizmu, elementem neutralnym działania jest graf K_1 .

Z łączności działania możemy zapisać $G_1 \square G_2 \square \dots \square G_k = G$ gdzie G jest produktem kartezjańskim grafów G_1, G_2, \dots, G_k , a następnie poetykietować wierzchołki grafu G k -elementową listą (v_1, v_2, \dots, v_k) gdzie $v_i \in V(G_i)$ dla $1 \leq i \leq k$. Jeżeli v etykietowany jest przez listę (v_1, v_2, \dots, v_k) , można zdefiniować rzutowanie $p_i : V \rightarrow V_i$ dla $1 \leq i \leq k$, które dane jest wzorem $p_i(v) = v_i$, gdzie v_i jest i -tym elementem listy etykietującej wierzchołek v . Wierzchołek v_i ten będzie i -tą współrzędną wierzchołka v .

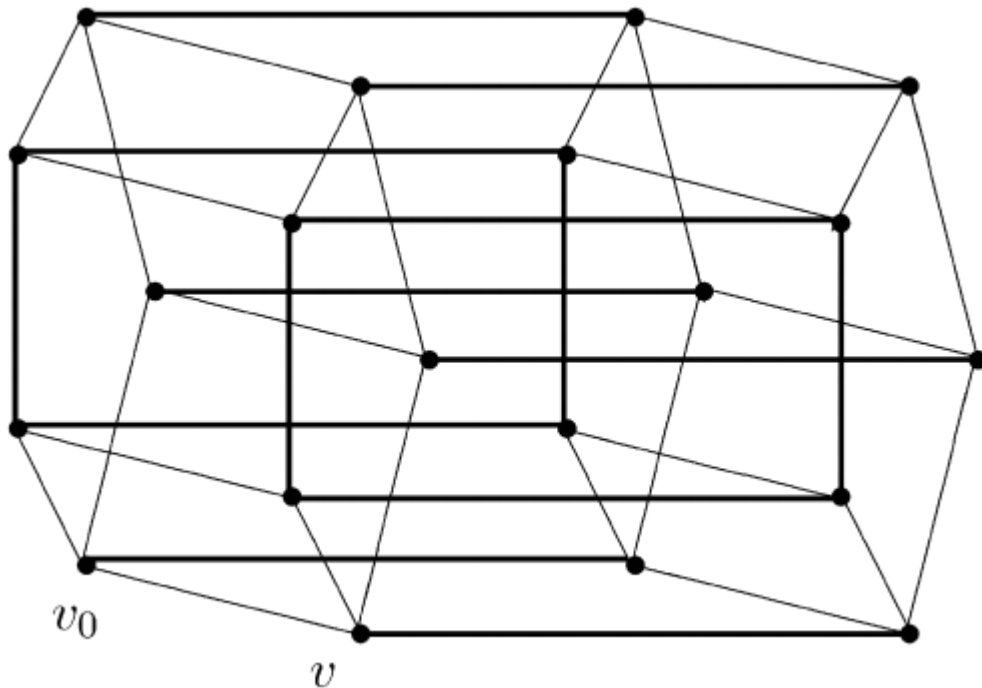
Jeżeli w grafie G dany jest wierzchołek v i rozważymy wierzchołki, które różnią się od wierzchołka v tylko na i -tej pozycji, to podgraf indukowany przez te wierzchołki utworzy graf izomorficzny z grafem G_i . Podgraf ten będzie nazywany i -tą warstwą G_i przechodzącą przez wierzchołek v a jego oznaczeniem będzie G_i^v .

Niech v_0 będzie wyróżnionym wierzchołkiem w grafie G . Warstwy przechodzące przez v_0 nazywamy warstwami jednostkowymi. Wierzchołek v_0 należy do każdej warstwy jednostkowej, natomiast zbiory $V(G_i^{v_0}) \setminus \{v_0\}$ są parami rozłączne dla $1 \leq i \leq k$.

Rysunek będzie zmieniony

2.2 Kolorowanie produktu iloczynu kartezjańskiego grafów

Niech dane będą dwa połączone wierzchołki u oraz v w grafie G . Analizując współrzędne tych wierzchołków, łatwo można stwierdzić że różnią się one dokładnie na jednej pozycji. Niech i oznacza tę pozycję. Wtedy krawędź uv należy do G_i^v . Krawędzi uv otrzymuje kolor i czyli $c(uv) = i$, gdzie c jest kolorowaniem właściwym produktu kartezjańskiego. Podsumowując: funkcja $c : E(G) \rightarrow 1, 2, \dots, k$ jest kolorowaniem właściwym produktu iloczynu kartezjańskiego jeżeli $c(uv) = i$ wtedy i tylko wtedy gdy współrzędne wierzchołków u oraz v różnią się na i -tej pozycji.



Każda krawędź należy dokładnie do jednej warstwy. Rozważając podgraf grafu G składający się z krawędzi koloru i to każda spójna składowa tego podgrafu będzie oddzielną i -tą warstwą grafu G .

2.3 Lemat o kwadracie

Niech graf G posiada właściwe pokolorowanie produktu kartezjańskiego. Dane są dwie połączone krawędzie e i f różnych kolorów. Wówczas istnieje dokładnie jeden kwadrat bez przekątnych (graf C_4) zawierający e oraz f .

Dowód:

Na początku rozważmy następujący fakt. Każdy trójkąt w grafie G jest pomalowany na ten sam kolor. Wynika to z tego, że dwa pierwsze wierzchołki różnią się na pozycji i -tej, natomiast jeżeli współrzędne trzeciego wierzchołka różniłyby się na pozycji j -tej która jest różna od i w porównaniu z pierwszym wierzchołkiem i jego współrzędnymi, nie byłoby możliwości aby drugi i trzeci wierzchołek byłyby ze sobą połączone, ponieważ ich współrzędne różniłyby się na dwóch pozycjach.

Na podstawie powyższego stwierdzenia stwierdzamy, że każdy kwadrat zawierający co najmniej jedną przekątną jest tego samego koloru.

Teraz właściwy dowód lematu. Dane są następujące oznaczenia:

$(v_0, v_1, \dots, v_i, \dots, v_j, \dots, v_k)$ współrzędne wspólnego wierzchołka v krawędzi e oraz f

$(v_0, v_1, \dots, v'_i, \dots, v_j, \dots, v_k)$ współrzędne wierzchołka v_e który jest drugim końcem krawędzi e

$(v_0, v_1, \dots, v_i, \dots, v'_j, \dots, v_k)$ współrzędnymi wierzchołka v_f który jest drugim końcem krawędzi f

v' wierzchołek o współrzędnych $(v_0, v_1, \dots, v'_i, \dots, v'_j, \dots, v_k)$

Łatwo stwierdzić, że wierzchołek v' jest połączony z wierzchołkiem v_e ponieważ ich współrzędne różnią się tylko na pozycji j oraz w grafie G_j istnieje krawędź $v_j v'_j$ ponieważ w grafie G istnieje krawędź f . Analogicznie stwierdzamy istnienie krawędzi $v' v_f$, co w połączeniu z faktem, że krawędzie e oraz f są różnego koloru i rozważaniom na temat kwadratów z przekątnymi daje nam tezę lematu. Co więcej na podstawie powyższego rozumowania, można wnioskować że przeciwległe krawędzie w kwadracie mają ten sam kolor, niezależnie czy kwadrat posiada przekątne czy też nie.

2.4 Lemat o izomorfizmie

Niech $G = (V, E)$ będzie spójnym grafem, natomiast E_1, E_2, \dots, E_k podziałem zbioru krawędzi. Niech każda spójna składowa $(V, \cup_{j \neq i} E_j)$ ma dokładnie jeden punkt wspólny z każdą spójną składową (V, E_i) oraz krawędzie między dwoma składowymi (V, E_i) wyznaczają izomorfizm między tymi składowymi (jeżeli takie krawędzie istnieją). Wtedy:

$$G = \Pi G_i$$

gdzie G_i jest dowolną, spójną składową (V, E_i) .

Dowód:

Tutaj chyba dowód indukcyjny ze względu na k , do sprawdzenia.

2.5 Lemat o udoskonaleniu faktoryzacji iloczynu kartezjańskiego

3 Faktoryzacja z dodatkowymi informacjami

W tym rozdziale przedstawimy algorytm kolorowania właściwego grafu względem iloczynu kartezjańskiego mając podane kolory krawędzi wychodzących z pewnego wierzchołka, względem danego rozkładu grafu. Następnie pokażemy jak mając kolory wszystkich krawędzi nadać współrzędne wszystkim wierzchołkom.

3.1 Algorytm kolorowania krawędzi

Załóżmy, że mamy dane kolory wszystkich krawędzi, w kolorowaniu iloczynu kartezjańskiego grafu, wychodzących z pewnego wierzchołka v_0 . Kolorowanie pozostałych krawędzi będzie odbywało się w kolejności przeszukiwania grafu w algorytmie BFS z wierzchołkiem początkowym v_0 .

Twierdzenie 3.1.1 Niech $G=G_1 \square G_2 \square \dots \square G_k$ będzie grafem spójnym. Dane jest kolorowanie właściwe względem podanego rozkładu dla wszystkich krawędzi wychodzących z pewnego wierzchołka v_0 . Wtedy kolorowanie właściwe produktu kartezjańskiego może być uzyskane zgodnie z kolejnością algorytmu BFS o wierzchołku początkowym v_0 . Złożoność czasowa tego algorytmu to $O(mn)$, natomiast złożoność pamięciowa $O(n^2)$.

Dowód

Jako dowód przedstawiony zostanie algorytm kolorowania krawędzi. W pierwszym kroku algorytmu dzielimy zbiór wierzchołków grafu G na podzbiory $L_0, L_1, L_2, \dots, L_r$ w taki sposób, że wierzchołek v należy do zbioru L_i wtedy i tylko wtedy gdy odległość wierzchołka v od wierzchołka v_0 jest równa i . Zbiory te będziemy nazywać poziomami. Następnie dla każdego wierzchołka, wszystkie krawędzie incydentne z tym wierzchołkiem dzielimy na trzy zbiory- zbiór krawędzi dolnych, poprzecznych i górnych. Definiowanie tych zbiorów przebiega następująco. Rozważy poziom i , następnie dla wszystkich wierzchołków v należących do zbioru L_i rozważamy krawędzie vu incydentne z v . Wówczas jeżeli u należy do L_{i-1} to krawędź vu będzie krawędzią dolną. Jeżeli u należy do L_i wówczas uv będzie krawędzią poprzeczną, jeżeli natomiast u należy do L_{i+1} wówczas uv będzie krawędzią górną wierzchołka warstwy L_i . Zauważmy, że krawędzie dolne wierzchołków poziomu L_{i+1} są krawędziami górnymi wierzchołków poziomu L_i .

Nasz algorytm rozpoczynamy od pokolorowania krawędzi poprzecznych L_1 . Nie stanowi to problemu, ponieważ każdy trójkąt jest monochromatyczny. Tak więc każdej krawędzi uv nadajemy kolor krawędzi v_0v czyli $c(uv) := c(vv_0) = c(uv_0)$.

Następnie indukcyjnie kolorujemy krawędzie dolne a następnie poprzeczne L_{i+1}

mając już pokolorowane krawędzie dolne i poprzeczne L_i . Nie ma potrzeby kolorowania krawędzi górnych L_i ponieważ zbiór ten jest również zbiorem krawędzi dolnych L_{i+1} .

Zaczynamy od krawędzi dolnych. Przeglądamy wierzchołki należące do L_{i+1} zgodnie z kolejnością wyznaczoną przez algorytm BFS. Niech dany będzie wierzchołek u oraz krawędź uv . Ponieważ wierzchołek v należy do L_i , gdzie $i \geq 1$ to istnieje wierzchołek w należący do L_{i-1} sąsiedni z v . Rozważmy dwa przypadki:

1. Nie istnieje wspólny sąsiad wierzchołków u oraz w różny od v . Wówczas nie istnieje kwadrat zawierający wierzchołki u oraz w a co za tym idzie kolory krawędzi uv oraz vw są te same czyli $c(uv) := c(vw)$.
2. Istnieje wspólny sąsiad x wierzchołków u oraz w różny od v . W tym przypadku $c(uv) := c(xw)$ oraz $c(ux) := c(vw)$.

Uzasadnienia w obydwu przypadkach wynikają z lematu o kwadracie (2.3).

Rozważmy teraz krawędzie poprzeczne L_{i+1} . W tym celu również przeglądamy wierzchołki należące do tego poziomu. Dla każdej krawędzi uv należącej do krawędzi poprzecznych rozważanego poziomu szukamy krawędzi dolnej uw i podobnie jak dla krawędzi dolnych szukamy wspólnego sąsiada wierzchołków v oraz w . Jeśli takowy wierzchołek x istnieje wówczas $c(uv) := c(wx)$, jeśli nie $c(uv) := c(uw)$.

Zauważmy, że aby wyznaczyć G_i wystarczy znaleźć $G_i^{v_0}$. Wierzchołek v należy do $V(G_i^{v_0})$ wtedy i tylko wtedy gdy wszystkie jego krawędzie dolne są koloru i . Tak więc aby wyznaczyć G_i wystarczy przejrzeć wszystkie krawędzie dolne wszystkich wierzchołków, jeżeli lista ta jest monochromatyczna wierzchołek ten będzie należał do $V(G_i^{v_0})$ gdzie i to kolor krawędzi dolnych tego wierzchołka.

W rozdziale 2.1 zdefiniowaliśmy warstwy jednostkowe $G_i^{v_0}$. Wierzchołki należące do $G_i^{v_0}$ będziemy nazywać wierzchołkami warstwy jednostkowej. Oczywiście wszystkie wierzchołki należące do L_1 będą wierzchołkami warstw jednostkowych. Rozważając jeszcze raz krawędzie dolne oraz poprzeczne wierzchołków warstw jednostkowych, na podstawie spójności produktu iloczynu kartezjańskiego stwierdzamy, że i krawędzie dolne i krawędzie górne tychże wierzchołków należą do warstw jednostkowych.

Na koniec pozostaje nam wykazanie, że nasz algorytm rzeczywiście spełnia założenia dotyczące złożoności pamięciowej i czasowej. Zauważmy że dla każdego wierzchołka należącego do L_i , gdzie $i > 0$ szukamy dolnego sąsiada, następnie przeglądamy wszystkie krawędzie dolne oraz poprzeczne. Tak więc wykonujemy co najwyżej $2m$ kroków w naszym algorytmie, gdzie m oznacza liczbę krawędzi naszego grafu G . Dla ustalonych krawędzi uv oraz uw szukamy wspólnego sąsiada x . Mamy co najwyżej $n = G(V)$ możliwości wyboru tego sąsiada. Jeżeli informacje o krawędziach grafu są przetrzymywane w tablicy sąsiedztwa sprawdzenie czy

dany wierzchołek jest sąsiadem innego można wykonać w czasie stałym. Tak więc dowiedliśmy, że złożoność czasowa algorytmu to $O(mn)$, natomiast złożoność pamięciowa $O(n^2)$.

3.2 Nadawanie współrzędnych wierzchołkom

W poprzednim podrozdziale opisaliśmy algorytm kolorowania krawędzi, jednakże nie podawaliśmy sposobu, jak nadać współrzędne wierzchołkom. Zdefiniujemy teraz algorytm, który nada współrzędne naszym wierzchołkom, mając już dane kolory wszystkich krawędzi.

Twierdzenie 3.2.1 Niech $G=G_1 \square G_2 \square \dots \square G_k$ będzie grafem spójnym. Dane jest również kolorowanie właściwe produktu względem podanego rozkładu. Algorytm nadania współrzędnych wierzchołkom może być zrealizowany w złożoności czasowej i pamięciowej $O(m)$.

Dowód

Dla każdego wierzchołka, aby móc przechować informację o jego współrzędnych, potrzebujemy k -elementowej tablicy. Ponieważ k jest mniejsze od minimalnego stopnia grafu oraz $\delta(G) \cdot n \leq m$ to całkowity rozmiar tablic ze współrzędnymi jest mniejszy bądź równy $O(m)$.

Algorytm rozpoczynamy od nadania wierzchołkowi v_0 współrzędnych składających się z samych 0. Następnie przeszukujemy wszystkie wierzchołki zgodnie z kolejnością algorytmu BFS.

Jeżeli wierzchołek należy do i -tej warstwy jednostkowej jego wszystkie współrzędne otrzymują wartość 0 z wyłączeniem i -tej współrzędnej. Przeszukując wierzchołki i -tej warstwy jednostkowej i -tej współrzędnej nadajemy kolejną liczbę naturalną. Zapisując formalnie jeżeli nasz wierzchołek u należy do warstwy jednostkowej $u_j = 0$ dla $j \neq i$ oraz $u_i = \max\{v_i\} + 1$ gdzie v_i to i -te współrzędne wierzchołków należących do G_i odwiedzonych wcześniej niż wierzchołek u w algorytmie BFS. Jeżeli natomiast wierzchołek nie należy do warstwy jednostkowej to istnieją co najmniej dwie krawędzie dolne tego wierzchołka, mające różne kolory. Niech tym wierzchołkiem będzie u natomiast jego krawędziami dolnymi uv oraz uw . Wówczas $u_i = \max(v_i, w_i)$ dla $1 \leq i \leq k$.

4 Etykietowanie produktu kartezjańskiego

W tym rozdziale rozszerzymy definicję kolorowania i nazwiemy ją etykietowaniem. Wszystkie krawędzie, rzutowane na tę samą krawędź otrzymają tę samą etykietę.

Pokażemy, że produkt może być poetykietowany, a co za tym idzie pokolorowany w czasie liniowym. Etykietowanie pozwoli nam na określenie pozycji danej krawędzi w stosunku do faktoryzacji iloczynu kartezjańskiego grafu w czasie stałym. W rozdziale tym opiszemy struktury danych niezbędne do etykietowania krawędzi. Właściwie etykietowanie okaże się pozycją krawędzi w tablicy krawędzi wychodzących z danego wierzchołka o tym samym kolorze.

Zauważmy, że używamy współrzędnych wierzchołka do określenia jego pozycji w produkcie iloczynu kartezjańskiego. Całkowita długość tych wektorów jest równa $O(m)$. Przez pozycję krawędzi uv rozumiemy pozycję wierzchołka u , kolor krawędzi uv oraz rzutowanie $p_i(uv)$ czyli $p_i(u)p_i(v)$ czyli bazę krawędzi uv .

Krawędź uv ma ten sam kolor co krawędź $p_i(u)p_i(v)$ oraz tak samo jest krawędzią dolną, poprzeczną lub górną. Poniżej przedstawimy jak efektywnie przetrzymywać informację o bazie.

W poprzednich rozdziałach opisywaliśmy w jaki sposób dzielimy krawędzie incydentne z danym wierzchołkiem na krawędzie dolne, poprzeczne i górne. Następnie każdą taką listę można podzielić na krawędzie tego samego koloru. Pozycja krawędzi, w tak stworzonej monochromatycznej liście, posłuży nam do zlokalizowania $p_i(uv)$, będziemy to nazywać numerem danej krawędzi o oznaczając $n(uv)$. W ogólnym przypadku $n(uv) \neq n(vu)$. Parę $\langle c(uv), n(uv) \rangle$ będziemy nazywać etykietą uv i oznaczać $l(uv)$. Razem z $p_{c(uv)}(u)$ będzie to opisywać pozycję krawędzi w produkcie. Pozycje krawędzi wychodzących z wierzchołka v_0 będą przechowywane w tablicy o długości $d(v_0)$, tak więc dostęp do nich można uzyskać w czasie stałym.

Tak więc etykietowanie zostało zdefiniowane. Jest ono zależne od kolejności krawędzi należących do warstw jednostkowych. Kolejność pozostałych krawędzi w listach monochromatycznych nie będzie miała wpływu na etykietowanie. Głównym celem naszego algorytmu będzie właśnie etykietowanie tychże krawędzi i zmiana numeru danej krawędzi w liście.

W algorytmie dobrze uwzględnić że krawędź dolna, poprzeczna lub górna uv ma początek w u natomiast koniec w v . Pozwoli to na znalezienie tej krawędzi w liście monochromatycznej w czasie stałym, jeżeli znamy jej numer. Zmodyfikujemy również macierz sąsiedztwa tak, aby w komórce uv mieć numer danej krawędzi właśnie w celu znalezienia jej numeru w czasie stałym.

Tak więc podsumowując w naszym algorytmie będziemy używać następujących struktur danych: dla każdego wierzchołka lista sąsiedztwa, zmodyfikowaną listę krawędzi oraz zmodyfikowaną tablicę sąsiedztwa. Dodatkowo każdy wierzchołek zostanie ułożony w tablicy zgodnie z kolejnością algorytmu BFS, będzie on posiadał numer swojego poziomu, wektor współrzędnych (o długości nie większej niż $d(v_0)$) oraz listę krawędzi dolnych, poprzecznych oraz górnych, które następnie będą dzielone na listy monochromatyczne. Budowa tychże struktur, z wyłączeniem list monochromatycznych jest możliwa w czasie $O(m)$. Pokażemy, że i te tablice

można zbudować w takim czasie. W następnym rozdziale pokażemy również, że tablica sąsiedztwa może być zbudowana z użyciem $O(m)$ pamięciowym.

Twierdzenie Niech $G = G_1 \square G_2 \square \dots \square G_k$ będzie grafem spójnym. Dane jest również kolorowanie właściwe produktu względem podanego rozkładu dla krawędzi wychodzących z wierzchołka v_0 . Etykietowanie tego grafu może być zrobione w czasie $O(m)$

Dowód

Opiszemy liniowy algorytm etykietowania. Poetykietowanie krawędzi wychodzących z v_0 nie stanowi problemu. Następnie etykietujemy krawędzie dolne i poprzeczne wierzchołków należących do warstwy L_1 . To również nie stanowi problemu ponieważ wszystkie krawędzie dolne dostają numer 1 natomiast krawędzie poprzeczne kolorujemy tak jak w rozdziale 3, numery tych krawędzi są zgodne z kolejnością w jakiej zostały podane na wejściu.

Zakładamy, że mamy poetykietowane krawędzie dolne i poprzeczne dla warstwy L_i i indukcyjnie etykietujemy krawędzie dolne i poprzeczne warstwy L_{i+1} . Niech wierzchołek u należy do warstwy L_{i+1} . Rozważamy następujące przypadki:

1. Wierzchołek ma tylko jedną krawędź dolną. Wówczas kolorujemy tak, jak to było w rozdziale 3.1 natomiast krawędź otrzymuje numer 1.
2. Szukamy kwadratu bazowego dla wierzchołków posiadających więcej niż jedną krawędź dolną.
 - a) Niech uv będzie pierwszą taką krawędzią. Natomiast niech vx będzie krawędzią dolną wierzchołka v . Szukamy wspólnego sąsiada wierzchołków u oraz x różnego oczywiście od wierzchołka v .
 Jeżeli wspólny sąsiad nie istnieje, wówczas $c(uv) := c(vx)$. Pozostałe krawędzie dolne również kolorujemy tym samym kolorem. Oznacza to, że wierzchołek u należy do warstwy jednostkowej, a co za tym idzie możemy ponumerować krawędzie zgodnie z tym, w jaki sposób podano je na wejściu.
 Tak więc rozważmy przypadek, że istnieje wierzchołek w będący wspólnym sąsiadem wierzchołków u oraz x . Jeżeli kolory krawędzi vx oraz vw są różne wówczas $l(uv) := l(wx)$ oraz $l(uw) := l(vx)$.
 - b) hg

4.1 Etykietowanie produktu w czasie liniowym

4.2 Sprawdzanie spójności kolorowanie właściwego produktu iloczynu kartezjańskiego

5 Opis algorytmu faktoryzacji

6 Implementacja Algorytmu

6.1 Wprowadzenie

6.2 Dane wejściowe

6.3 Opis pakietów i ważniejszych klas

6.4 Opis algorytm faktoryzacji