

Krzysztof Giaro*, Marek Kubale*

Szeregowanie rozrzedzonych systemów zadań jednostkowych 1- i 2-procesorowych w oknach czasowych

1. Wprowadzenie

Rozważmy zbiór niepodzielnych zadań o jednostkowych czasach wykonania, z których każde wykorzystuje co najwyżej dwa przypisane mu procesory dedykowane, zaś tylko niektóre chwile czasu ze względów zasobowych lub logistycznych mogą być dostępne dla określonych zadań. Żadna maszyna nie może też pracować nad więcej niż jednym zadaniem równocześnie. Tego typu model znajduje zastosowania wszędzie tam, gdzie poszczególne zadania wymagają współpracy nie więcej niż dwóch podmiotów, a więc np. przy przesyłaniu plików ze stacji źródłowych do docelowych [1] lub w szeregowaniu testów komputerowych [6].

Sformalizujmy nasze zagadnienie.

Instancja problemu składa się ze skończonych zbiorów:

- niepodzielnych zadań $T = \{T_1, T_2, \dots\}$ o jednostkowym czasie wykonywania,
- maszyn (zwanych też procesorami) $M = \{M_1, M_2, \dots\}$,

przy czym każdemu zadaniu T_j odpowiada jedno- lub dwuelementowy podzbiór $fix_j \subseteq M$ określający procesory wykorzystywane w sposób równoległy przez to zadanie. Ponadto dana jest funkcja $L: T \rightarrow 2^N$ przypisująca każdemu zadaniu T_i skończony zbiór $L(T_i)$ chwil czasu, w których może się ono wykonywać. Poprawnym harmonogramem nazwiemy takie przypisanie zadaniom jednostek czasu, numerowanych kolejnymi liczbami naturalnymi, w którym żadne dwa zadania nie używają jednocześnie tego samego procesora, a każde zadanie wykonuje się w jednej z dostępnych mu chwil.

W dalszej analizie pomocny okaże się następujący model grafowy. Możemy przyjąć, że procesory stanowią wierzchołki multigrafu, którego krawędziami są szeregowane zadania: każde zadanie dwuprocesorowe łączy ze sobą oba wykorzystywane procesory, zaś jednoprocesorowe stanowi pętlę w grafie przy odpowiednim procesorze. Wygodniej jednak jest rozważać multigrafy bez pętli i tak też postąpimy dalej – wystarczy zatem dodać dla każdego zadania jednomaszynowego nowy fikcyjny „drugi procesor”, a wtedy w multigrafie utworzonym zgodnie z powyższymi regułami pętle zostają zastąpione przez krawędzie wiszące.

* Politechnika Gdańska, Wydział ETI, Katedra Algorytmów i Modelowania Systemów

Łatwo zauważyć, że jeśli potraktować numer chwili czasu, w której wykonuje się zadanie, jak kolor przypisany odpowiadającej mu krawędzi grafu, wówczas warunki poprawności harmonogramu pokrywają się z wymogami legalności krawędziowego kolorowania listowego. Rzeczywiście, fakt, że żadne dwa zadania współdzielące procesor nie mogą wykonywać się równocześnie, oznacza, że krawędzie spotykające się przy wierzchołku muszą mieć różne kolory. Natomiast ograniczona dostępność zadań w czasie określona za pomocą funkcji L sprowadza się do tego, że kolor każdej krawędzi e musi należeć do jej listy $L(e)$. Możemy więc sprowadzić zagadnienie istnienia poprawnego harmonogramu w naszym modelu do listowego kolorowania krawędziowego multigrafów.

W szeregowaniu zadań zasadnicze znaczenie ma również kwestia znalezienia uszeregowania optymalnego względem różnych kryteriów kosztu. Najczęściej spotykamy funkcje kryterialne dwojakiego typu.

Pierwszy rodzaj, zawierający m.in. klasyczne C_{\max} , T_{\max} , F_{\max} definiowany jest za pomocą zbioru funkcji (najczęściej niemalejących, począwszy od pewnego argumentu) $f_{T_i}: L(T_i) \rightarrow N \cup \{0\}$, gdzie $T_i \in T$ – zakładamy, że koszt wynikający z zakończenia wykonywania zadania T_i w chwili C_i wynosi $f_{T_i}(C_i)$, poszukujemy zaś uszeregowania minimalizującego największy z tych kosztów, tj. wyrażenie $f_{\max} = \max \{f_{T_i}(C_i): T_i \in T\}$. Problem szeregowania z tak określonym kryterium optymalizacyjnym nadal można łatwo sprowadzić do listowego kolorowania krawędziowego: wystarczy dla zmiennej l o wartości początkowej $\max_{T_i \in T, j \in L(T_i)} f_{T_i}(j)$ znaleźć np. metodą szukania binarnego najmniejszą wartość, dla której graf opisujący system można jeszcze pokolorować listowo przy zestawie list $L'(T_i) = f_{T_i}^{-1}(\{0, \dots, l\})$. Jest oczywiste, że optymalny koszt harmonogramu f_{\max} jest równy tejże minimalnej wartości l , a odpowiadające jej pokolorowanie przedstawia szukany harmonogram.

Druga grupa kryteriów m.in. takich jak ΣC_j , $\Sigma w_j T_j$, $\Sigma w_j U_j$ bazuje na funkcjach kosztu uszeregowania wyznaczanych według formuły $\sum_{T_i \in T} f_{T_i}(C_i)$. W tym wypadku model grafowy musimy rozszerzyć o przypisane krawędziom funkcje f_{T_i} , zaś optymalne uszeregowanie opisane jest listowym pokolorowaniem krawędziowym minimalizującym sumę kosztów użytych barw. Taki model kolorowania stanowi uogólnienie znanych w teorii grafów pokolorowań zarówno listowych, jak i sumacyjnych.

Pragniemy znaleźć wielomianowy algorytm optymalnie szeregujący zadania względem rozmaitych funkcji kryterialnych i dla możliwie szerokiej klasy systemów. Niestety, znane dotąd wyniki wskazują na NP -trudność zagadnienia nawet przy bardzo prostych klasycznych kryteriach i w sytuacji, gdy zadania są zawsze dostępne. Na przykład powszechnie wiadomo, że obliczeniowo trudne są już $P | \text{fix}_j = 2, p_j = 1 | C_{\max}$ i $P | \text{fix}_j = 2, p_j = 1 | \Sigma C_j$. Skoro szukamy efektywnych i dokładnych algorytmów – konieczne będzie narzucenie dalszych ograniczeń na strukturę rozważanego zestawu zadań. Pokażemy, że warunkiem umożliwiającym istnienie efektywnego algorytmu jest ograniczona liczba cykli w grafie systemu.

2. Systemy o grafach acyklicznych

Przyjmijmy następującą notację: $G(V, E)$ oznacza multigraf o zbiorze wierzchołków V i krawędzi E , które w przypadku grafów prostych będą traktowane jak nieuporządkowane pary wierzchołków, tj. krawędź łącząca u z v oznaczmy przez $\{u, v\}$. Stopień wierzchołka v

to $\deg(v)$, tradycyjnie maksymalny stopień wierzchołka w multigrafie oznaczmy przez Δ . W sytuacji gdy ze względu na większą liczbę rozważanych grafów oznaczenia powyższe mogłyby okazać się niejednoznaczne, po stosownym symbolu pojawi się symbol grafu, którego dane pojęcie dotyczy np. $\deg_G(v)$ lub $V(G)$. Rodzinę wszystkich skończonych podzbiorów zbioru A oznaczmy przez $P_{fin}(A)$. Wreszcie N będzie zbiorem liczb naturalnych. Podamy najpierw formalne definicje opisanych wyżej modeli kolorowania.

Definicja 1

Niech dany będzie multigraf $G(V,E)$. Funkcję $c: E \rightarrow N$ nazywamy pokolorowaniem krawędziowym wtedy i tylko wtedy, gdy dla każdej pary krawędzi sąsiednich e, f zachodzi $c(e) \neq c(f)$.

Definicja 2

Dla grafu $G(V,E)$ i zestawu list $L: E \rightarrow P_{fin}(N)$ pokolorowanie krawędziowe c nazwiemy L -listowym, gdy $\forall_{e \in E} c(e) \in L(e)$. Jeżeli dodatkowo znane są funkcje kosztów $f_e: L(e) \rightarrow N \cup \{0\}$ dla wszystkich $e \in E$, wówczas kosztem pokolorowania c nazwiemy sumę $\sum_{e \in E} f_e(c(e))$. Pokolorowanie L -listowe minimalizujące ten koszt jest optymalnym listowo-kosztowym pokolorowaniem krawędziowym.

W dalszym ciągu pomocne okażą się następujące fakty:

Definicja 3

Skojarzeniem w grafie $G(V,E)$ nazwiemy dowolny podzbiór $B \subseteq E$ krawędzi nie sąsiadujących ze sobą. Skojarzenie jest pełne lub doskonałe, gdy $|B| = |V|/2$.

Lemat 1

([9]) Największe skojarzenie w grafie $G(V,E)$ można znaleźć w czasie $O(|E||V|^{1/2})$.

Lemat 2 ([5])

Dla danego grafu dwudzielnego $G(V,E)$ o krawędziach obciążonych wagami $w: E \rightarrow N \cup \{0\}$ można znaleźć najcięższe (w sensie sumy wag) skojarzenia we wszystkich podgrafach G indukowanych przez zbiory $V - \{v\}$ dla $v \in V$ w czasie $O(|E||V|^{1/2} \log(|V| \max_{e \in E} w(e)))$.

W tym samym czasie znajdujemy również wszystkie najlżejsze skojarzenia o maksymalnej możliwej mocy w tych podgrafach – wystarczy zastosować powyższy algorytm po zmodyfikowaniu wag, np. według formuły $w'(e) = \lfloor |V| \max_{e \in E} w(e)/2 \rfloor - w(e) + 1$.

Definicja 4

Dla grafu $G(V,E)$ zestaw list $L: E \rightarrow P_{fin}(N)$ jest dokładny, gdy:

$$\forall_{\{u,v\} \in E} |L(\{u,v\})| = \deg(u) + \deg(v) - 1.$$

Lemat 3

Ogólną wersję zagadnienia listowo-kosztowego kolorowania grafów można wielomianowo zredukować do tegoż problemu z dodatkowym warunkiem dokładności zestawu list.

Dowód. Niech C będzie największym z kosztów kolorów występujących na wszystkich listach w zestawie L dla pewnego grafu $G(V,E)$. Tworzymy zestaw L' list, uzupełniając listy zbyt krótkie (w sensie definicji 4) o nowe, unikalne, tj. nie występujące w innych zbiorach barwy, przypisując im jednocześnie koszt $|E|C + 1$. Wtedy istnienie pokolorowania L -listowego jest równoważne temu, że optymalne pokolorowanie L' -listowe ma koszt mniejszy od $|E|C+1$. Następnie listy „zbyt długie” możemy skrócić, usuwając elementy o największych kosztach, nie zmieniając optymalnego kosztu pokolorowania L' -listowego, gdyż dla dokładnego zestawu list każde listowe pokolorowanie częściowe można przedłużyć na wszystkie krawędzie. ■

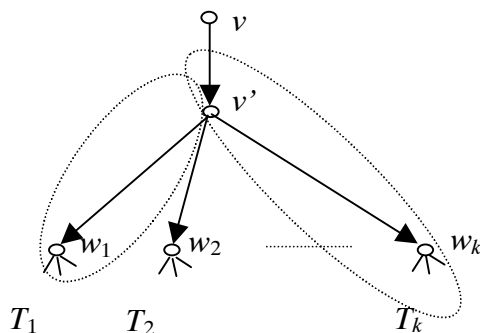
Tak więc w dalszych rozważaniach bez zmniejszenia ogólności będziemy mogli ograniczyć się do problemu kosztowego kolorowania listowego z dokładnymi listami.

Twierdzenie 1

Optymalne kosztowe L -listowe pokolorowanie krawędziowe drzewa $G(V,E)$ można znaleźć w czasie $O(|V|\Delta^2 \log(|V|C))$, gdzie $C = \max_{e \in E, i \in L(e)} f_e(i)$.

Dowód. Zaprezentujemy część procedury, umożliwiającą określenie minimalnej wartości kosztu pokolorowania – rozszerzenie jej o funkcję kolorującą graf jest proste. Przyjmujemy, że zestaw list L jest dokładny dla G . Dla pary wierzchołków sąsiednich v i v' oznaczymy przez H największe poddrzewo G , które zawiera krawędź $\{v, v'\}$, a v jest w nim wierzchołkiem wiszącym. Funkcja $\text{Val}(v, v'): L(\{v, v'\}) \rightarrow N \cup \{0\}$ będzie określona tak, że $\text{Val}(v, v')(i)$ jest minimalnym kosztem pokolorowania $L_{|E(H)}$ -listowego, krawędziowego drzewa H (z funkcjami kosztów odziedziczonymi z G), w którym krawędź $\{v, v'\}$ uzyskała barwę i . Rekurencyjnie wyznaczmy wartości funkcji $\text{Val}(u, u')$ dla pewnego wierzchołka wiszącego u w G , wówczas szukany optymalny koszt będzie równy $\min_{i \in L(\{u, u'\})} \text{Val}(u, u')(i)$.

Jeżeli H jest krawędzią $\{v, v'\}$, to oczywiście $\text{Val}(v, v') = f_{\{v, v'\}}$. Załóżmy więc, że na dalszym etapie procedury mamy poddrzewo H jak na rysunku (rys. 1), przy czym znane są już funkcje $\text{Val}(v', w_l)$, $l = 1, \dots, k$.



Rys. 1. Drzewo H i jego poddrzewa

Aby wyznaczyć wartości funkcji $\text{Val}(v, v')$ konstruujemy graf dwudzielny $K^{v, v'}$, którego wierzchołki pierwszej partycji to $\{w_1, \dots, w_k\}$, drugą zaś tworzą liczby z $\bigcup_{l=1, \dots, k} L(\{w_l, v'\})$, przy czym istnieje krawędź $\{w_l, x\}$ wtedy i tylko wtedy, gdy $x \in L(w_l)$ i wówczas obciążamy ją wagą liczbową $\text{Val}(v', w_l)(x)$. Wtedy $\text{Val}(v, v')(j)$ jest równe wadze najlżejszego k -krawędziowego skojarzenia w grafie $K^{v, v'} - \{j\}$ (koszt pokolorowań krawędziowych poddrzew T_l) powiększonej o składnik $f_{\{v, v'\}}(j)$ związany z nadaniem krawędzi $\{v, v'\}$ barwy j .

Wyznaczenie funkcji $\text{Val}(v, v')$ wymaga obliczenia wag najlżejszych k -krawędziowych skojarzeń we wszystkich grafach $K^{v, v'} - \{j\}$ dla $j \in L(\{v, v'\})$. Wykorzystując algorytm z lematu 2 dla grafu $K^{v, v'}$ (zachodzi $|V(K^{v, v'})| = O(\deg_G(v')\Delta(G)) = |E(K^{v, v'})|$), wykonamy to w czasie $O(\deg_G(v')\Delta(G)^2 \log(|V(G)|\Delta(G)C))$. Oszacowanie złożoności całego algorytmu kolorowania drzewa uzyskamy sumując ten czas po wszystkich wierzchołkach v' , co daje nam $O(|V|\Delta^2 \log(|V|C))$. ■

Analogiczne twierdzenie w modelu bez list (każdej krawędzi można przypisać dowolny kolor) wykazywano już w [3] i [10]. Praca [4] podaje też podobny wynik dla tzw. kolorowania totalnego, tj. równocześnie wierzchołków i krawędzi. Można zastanowić się nad ogólniejszymi niż drzewa rodzinami grafów umożliwiającymi rozwiązanie problemu w zbliżony sposób. Okazuje się, że wielomianowe algorytmy dopuszczają wszystkie grafy o ograniczonej przez stałą liczbie cykli.

Definicja 5

Liczba cyklomatyczna *multigrafu* $G(V, E)$ wynosi $\gamma(G) = |E| - |V| + v(G)$, gdzie $v(G)$ jest liczbą składowych spójności.

Twierdzenie 2

Dla każdego ustalonego $k \in \mathbb{N}$ problem krawędziowego L -listowego kolorowania kosztowego dla grafów G o liczbie cyklomatycznej $\gamma(G) \leq k$ można rozwiązać w czasie $O(|V|\Delta^{2+k} \log(|V|C))$, gdzie $C = \max_{e \in E, i \in L(e)} f_e(i)$.

Dowód. Bez zmniejszenia ogólności można przyjąć, że $G(V, E)$ jest grafem spójnym o liczbie cyklomatycznej k , L jego dokładnym zestawem list, zaś $A = \{e_1, \dots, e_k\}$ zbiorem tych krawędzi, których usunięcie tworzy z G drzewo. Dalej niech U będzie zbiorem wszystkich wierzchołków incydentnych do krawędzi z A . Wystarczy skonstruować procedurę o złożoności $O(|V(G)|\Delta(G)^2 \log(|V(G)|C))$, która dla danego pokolorowania krawędziowego $L|_A$ -listowego grafu (U, A) znajdzie jego przedłużenie o minimalnym możliwym koszcie do krawędziowego pokolorowania L -listowego G . Wykorzystamy procedurę z twierdzenia 1.

Na podstawie grafu G , jego zestawu list i kosztów tworzymy drzewo D następująco:

1. usuwamy krawędzie z A ;
2. dla każdej takiej krawędzi $\{u, v\} \in A$ wprowadzamy dwie nowe krawędzie wiszące $\{u, u_{new}\}$, $\{v, v_{new}\}$ z listami singletonowymi postaci $L^D(\{v, v_{new}\}) = L^D(\{u, u_{new}\}) = \{d(u, v)\}$; ponadto definiujemy dla nich funkcje kosztów: jedną wzorem $f^D_{\{v, v_{new}\}} = f_{\{v, u\}}$, a drugą $f^D_{\{u, u_{new}\}} = 0$;
3. list ani kosztów pozostałych krawędzi grafu nie zmieniamy.

Optymalny koszt listowego pokolorowania krawędziowego drzewa D i przedłużenia funkcji d do pokolorowania kosztowego całego G są równe – krawędzie $\{u, u_{new}\}$ i $\{v, v_{new}\}$ odpowiadają połówkom „rozciętej” krawędzi $\{u, v\}$, zaś jednoelementowe listy tych krawędzi „wymuszają”, by kosztowo optymalne pokolorowanie D przedłużało d . Wystarczy już tylko odnotować, że $|V(D)| = |V(G)| + 2k$ oraz $\Delta(D) = \Delta(G)$. ■

Oczywiście przedstawioną procedurę można stosować również dla multigrafów. Rozszerzając notację trójpolową o symbol *win*, sygnalizujący istnienie okien czasowych przypisanych zadaniom, oraz oznaczając przez G multigraf opisujący badany system: możemy stwierdzić, że:

Twierdzenie 3

Dla każdego ustalonego $k \in N$ problemy $P|win, fix_j \leq 2, p_j = 1, \gamma(G) \leq k|$ -, $P|win, fix_j \leq 2, p_j = 1, \gamma(G) \leq k|f_{max}$ oraz $P|win, fix_j \leq 2, p_j = 1, \gamma(G) \leq k|\Sigma f_j$ są wielomianowe, gdzie występujący w kryteriach symbol f_j odpowiada funkcjom czasu zakończenia zadania T_j .

3. Multikolorowanie a szeregowanie zadań

Okazuje się, że samo tylko dodanie do drzewa dodatkowych krawędzi równoległych sprawia, iż rozważane problemy kolorowania stają się trudnymi obliczeniowo.

Definicja 6

Szkieletem multigrafu G nazywamy graf prosty powstający z niego wskutek sukcesywnego usuwania jedna po drugiej krawędzi, dla których istnieją jeszcze inne krawędzie równoległe (tzn. o tych samych obu wierzchołkach końcowych). Multigraf którego szkieletem jest drzewo (odpowiednio: ścieżka, las) to multidrzewo (multiścieżka, multilas).

Twierdzenie 4

Problem istnienia L -listowego multipokolorowania krawędziowego multiścieżki jest NP-zupełny.

Dowód. Przeprowadzimy redukcję z problemu 3-spełnialności. Niech ϕ będzie formułą logiczną w postaci 3CNF, tj. koniunkcją zbioru klauzul $C = \{c_1, \dots, c_m\}$, przy czym $X = \{x_1, \dots, x_n\}$ to zbiór występujących w niej zmiennych logicznych, zaś $X^* = \{x, \sim x: x \in X\}$ zawiera wszystkie literały, które mogą pojawić się w ϕ . Każda klauzula jest alternatywą trzech literałów z X^* , można też przyjąć, że żadna zmienna logiczna nie występuje w tej samej klauzuli więcej niż raz. Skonstruujemy multiścieżkę F o szkielecie P_4 , która posiada listowe pokolorowanie krawędziowe wtedy i tylko wtedy, gdy istnieje podstawienie wartości logicznych 0 i 1 do zmiennych z X nadające całej ϕ wartość 1.

Weźmy dowolną różnowartościową funkcję $g: C \times X^* \rightarrow N$. Dla każdego $x \in X$ utworzymy najpierw graf pomocniczy F_x będący multidrzewem o szkielecie w postaci ścieżki długości 2 łączącej kolejne wierzchołki u_x, v_x, w_x . Składa się on $2m$ krawędzi równoległych: u_x z v_x łączą $e(c_i, x)$, a v_x z w_x łączą $f(c_i, x)$ dla $i = 1, \dots, m$. Ponadto przypisujemy listy dostępnych barw $L(e(c_i, x)) = \{g(c_i, x), g(c_i, \sim x)\}$ oraz $L(f(c_i, x)) = \{g(c_i, \sim x), g(c_{i+1}, x)\}$ – dla uprosz-

czenia zapisu przyjęliśmy tu $c_{m+1} = c_1$. Jest jasne, że istnieją tylko dwa pokolorowania listowe c grafu F_x :

- 1) $c(f(c_i, x)) = g(c_i, \sim x)$ i $c(e(c_i, x)) = g(c_i, x)$ dla $i = 1, \dots, m$ – wtedy na krawędziach przy wierzchołku u_x pozostają wolne kolory $g(c_i, \sim x)$;
- 2) $c(f(c_i, x)) = g(c_{i+1}, x)$ i $c(e(c_i, x)) = g(c_i, \sim x)$ dla $i = 1, \dots, m$ – wówczas przy wierzchołku u_x mamy wolne kolory $g(c_i, x)$.

Pierwsze z nich opowiada przypisaniu x fałszu (0), a drugie logicznej jedynce.

Cały multigraf F konstruujemy, sklejając oddzielne F_x dla $x \in X$, tj utożsamiając ze sobą wszystkie wierzchołki u_x (powstaje wspólny wierzchołek u), podobnie v_x razem dają v , a w_x łączą się w w . F jest więc multiścieżką łączącą kolejno u, v, w . Zauważmy, że ze względu na różnowartościowość g , żadne ze wspomnianych pokolorowań F_x i F_y dla $x \neq y$ nie kolidują ze sobą. Następnie dla każdej klauzuli $c_i \in C$ (przyjmijmy że stanowi ona alternatywę literałów $\alpha_i, \beta_i, \chi_i \in X^*$) doklejamy do u nową krawędź wiszącą $e(c_i)$ z listą $L(e(c_i)) = \{g(c_i, \alpha_i), g(c_i, \beta_i), g(c_i, \chi_i)\}$. Widzimy, że można dla niej wybrać jakiś kolor, jeżeli choć jedna barwa z $L(e(c_i))$ jest wolna przy wierzchołku u , co oznacza, że pokolorowania krawędzi grafów F_x odpowiadają przypisaniu jedynki logicznej przynajmniej jednemu z jej literałów. Na koniec utożsamimy drugie, tj. różne od u końce krawędzi $e(c_i)$ dla $c_i \in C$, tworząc z nich krawędzie równoległe. ■

Mimo to możliwe jest rozwiązanie analogicznego problemu kolorowania w wielomianowym czasie nawet dla multigrafów o dowolnie dużej liczbie cykloatycznej, jeżeli wielkość tego parametru jest jedynie skutkiem występowania *nierozróżnialnych* krawędzi równoległych (tzn. o identycznych parametrach $L(e)$ i f_e). W języku szeregowania zadań warunek ten oznacza, że dla zadań dwuprocessorowych okna czasowe i kryteria kosztu są funkcjami pary wykorzystywanych przez zadanie maszyn.

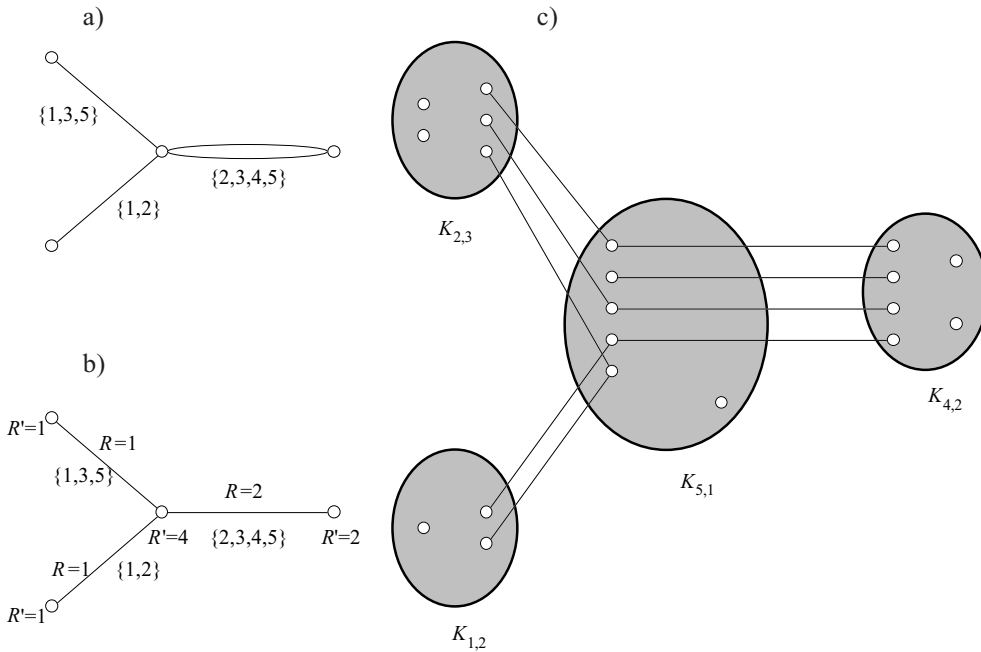
Korzystnie jest zreformułować to zagadnienie przedstawiając je jako tzw. *multikolorowanie szkieletu multigrafu*.

Definicja 7

Niech dany będzie graf prosty $G(V, E)$ i funkcja rozmiaru $R: E \rightarrow N$. Multipokolorowaniem krawędziowym nazwiemy funkcję $c: E \rightarrow P_{fin}(N)$ spełniającą warunki $\forall_{e \in E} |c(e)| = R(e)$ oraz dla każdej pary krawędzi sąsiednich e, f zachodzi $c(e) \cap c(f) = \emptyset$.

Jeżeli funkcja rozmiaru dla krawędzi e szkieletu określa liczbę krawędzi równoległych do e w pierwotnym multigrafie, przy czym wszystkim takim krawędziom przypisano jednokowe listy i ewentualnie funkcje kosztów, to multikolorowanie jest równoważne zwykłemu kolorowaniu tegoż multigrafu: zbiory przypisywane krawędziom szkieletu odpowiadają barwom wszystkich krawędzi do nich równoległych. Widać też, że zwykłe kolorowanie staje się szczególnym przypadkiem multikolorowania, w którym funkcja $R = 1$. Multikolorowanie listowe musi dodatkowo spełniać warunek poprawności $c(x) \subseteq L(x)$. Możemy również w naturalny sposób przedłużyć kryterium kosztu, określając je formułą $\sum_{e \in E} \sum_{i \in c(e)} f_e(i)$.

Wielomianowy algorytm rozwiązujący problem listowego (a więc bez funkcji kosztów) kolorowania krawędziowego multigrafów z nierozróżnialnymi krawędziami równoległymi (czyli o równych listach) podał Marx w [8] (rys. 2).



Rys. 2. Algorytm Marxa: a) multidrzewo; b) jego szkielet; c) graf D (szare elipsy oznaczają pełne grafy dwudzielne)

Twierdzenie 5 ([8])

Problem istnienia L -listowego multipokolorowania krawędziowego dla drzewa G i funkcji rozmiaru $R(G): E(G) \rightarrow N$ jest wielomianowy.

Naszkicujemy za [8] dowód twierdzenia. Dla grafu prostego $G(V,E)$ z funkcją $R: E \rightarrow N \cup \{0\}$ definiujemy inne odwzorowanie $R': V \rightarrow N \cup \{0\}$ dane wzorem

$$R'(v) = \sum_{e \in E} R(e)$$

Okazuje się, że w przypadku drzewa funkcja R' jednoznacznie wyznacza R . Pozwala to na sprowadzenie istnienia L -listowego multikolorowania krawędziowego drzewa do zagadnienia istnienia *zmodyfikowanego L -listowego multikolorowania c* grafu z warunkami poprawności określającymi nie liczbę barw przypisanych poszczególnym krawędziom, lecz spotykających się przy wierzchołkach:

$$\begin{aligned} \forall e \in E c(e) &\subseteq L(e) \\ \forall e, f \in E e \cap f \neq \emptyset &\Rightarrow c(e) \cap c(f) = \emptyset \\ \forall v \in V \sum_{e \in E} |c(e)| &= R'(v) \end{aligned} \quad (*)$$

To ostatnie zredukowano do wyszukiwania doskonałego skojarzenia. Tworzymy mianowicie graf D jako sumę rozłącznych kopii $G_i(V_i, E_i)$ grafu G dla wszystkich $i \in \bigcup_{e \in E} L(e)$ (odpowiednik elementu $x \in V(G) \cup E(G)$ w kopii G_i będziemy dalej oznaczać przez x_i). Z takiego grafu usuwamy wszystkie krawędzie e_i , takie że $i \notin L(e)$, a następnie wierzchołki izolowane. Chcemy, by doskonałe skojarzenia w D odpowiadały zmodyfikowanym multipokolorowaniom c (warunki (*)), przy czym przynależność krawędzi e_i do skojarzenia ma oznaczać, że kolor $i \in c(e)$. Musimy zatem sprawić, by dla każdego ustalonego $v \in V$ z wierzchołków v_i dla $i \in \bigcup_{e \in E} L(e)$ wychodziło łącznie dokładnie $R'(v)$ krawędzi skojarzenia należących do $\bigcup_{e \in E} E_i$. Efekt ten uzyskujemy, dodając do grafu D zbiór nowych wierzchołków V_v mocy $|\bigcup_{e \in E} L(e)| - R'(v)$, które wraz z wymienionymi v_i tworzą dwie partycje pełnego grafu dwudzielnego. Operację tę powtarzamy dla wszystkich $v \in V$. Łatwo zauważyć, że w uzyskanym D każde doskonałe skojarzenie (o ile takowe istnieje) opisuje pewne listowe L -multipokolorowanie drzewa G . Ale $|V(D)| = O(|V(G)|\Delta_L)$ oraz $|E(D)| = O(|V(G)|\Delta_L^2)$, gdzie oznaczyliśmy $\Delta_L = \max_{v \in V(G)} \sum_{e \in E(G)} |L(e)|$, więc korzystając z lematu 1 rozwiązujemy problem istnienia L -listowego krawędziowego multipokolorowania drzewa w czasie $O(|V(G)|^{1.5} \Delta_L^{2.5})$.

Metodę powyższą można dostosować do wyszukiwania optymalnego listowo-kosztowego pokolorowania krawędziowego multilasu.

Twierdzenie 6

Problem kosztowego L -listowego multikolorowania krawędziowego dla drzewa G i funkcji rozmiaru $R(G): E(G) \rightarrow N$ można rozwiązać w czasie wielomianowym.

Dowód. Obciążamy krawędzie grafu D z poprzedniego dowodu wagami w równymi kosztom odpowiadających im pokolorowań krawędzi z G , tj. $\forall_{e \in E(G)} \forall_{i \in L(e)} w(e_i) = f_e(i)$ (pozostałe krawędzie mają wagę 0), a następnie wyszukujemy najlżejsze pełne skojarzenie w D jednym z szybkich algorytmów np. tym z lematu 2. Znalezione skojarzenie w znany sposób opisuje najtańsze multipokolorowanie. ■

Rozszerzając symbolikę notacji trójpolowej o wyrażenie $G=class$, oznaczające przynależność grafu systemu do konkretnej rodziny multigrafów, możemy przepisać powyższe fakty w następującej postaci

Wniosek 1

Problem $P|win, fix_j \leq 2, p_j = 1, G = multipath|$ – jest NP-zupełny.

Wniosek 2 *Jeśli okna czasowe i funkcje kosztów dla zadań dwuprocesorowych wykorzystujących te same maszyny są równe, to problemy $P|win, fix_j \leq 2, p_j = 1, G = multiforest|$ –, $P|win, fix_j \leq 2, p_j = 1, G = multiforest|_{f_{max}}$ oraz $P|win, fix_j \leq 2, p_j = 1, G = multiforest|\Sigma f_j$ są wielomianowe.*

Nie wiadomo, czy wyniki te można uogólnić, podobnie jak w twierdzeniu 2 na multigrafy o szkieletach z niezerową lecz ograniczoną liczbą cykli, nie są bowiem znane odpowiednie deterministyczne procedury wielomianowe. Autor [8] sugeruje jednak pewną efektywną metodę rozwiązania wykorzystującą algorytmy probabilistyczne. Podamy bez dowodu następujące fakty.

Twierdzenie 7 ([8])

Dla każdego ustalonego $k \in \mathbb{N}$ problem istnienia L -listowego multipokolorowania krawędziowego dla grafu G o liczbie cyklomatycznej $\gamma(G) \leq k$ i funkcji rozmiaru $R(G): E(G) \rightarrow \mathbb{N}$ można rozwiązać wielomianowym algorytmem probabilistycznym.

Twierdzenie 8

Dla każdego ustalonego $k \in \mathbb{N}$ problem kosztowego L -listowego multikolorowania krawędziowego dla grafu $G(V, E)$ o liczbie cyklomatycznej $\gamma(G) \leq k$ i funkcji rozmiaru $R(G): E(G) \rightarrow \mathbb{N}$ można rozwiązać pseudowielomianowym algorytmem probabilistycznym.

Wyrażając ostatnie twierdzenie w języku szeregowania zadań, możemy stwierdzić, że problem szeregowania w oknach czasowych rozrzedzonych zadań jednostkowych k -procesorowych ($k \leq 2$) z powtórzeniami może być rozwiązany algorytmem probabilistycznym w czasie pseudowielomianowym.

Literatura

- [1] Coffman E. Jr., Garey M., Johnson D., LaPaugh A.: *Scheduling file transfers*. SIAM J. Computing, 14, (1985), 744–780
- [2] Giaro K., Kubale M.: *Edge-chromatic sum of trees and bounded cyclicity graphs*. Inf. Process. Lett., 75, (2000), 65–69
- [3] Giaro K., Kubale M., Piwakowski K.: *Complexity results on open shop scheduling to minimize total cost of operations*. Inter. J. Comp. Sys. Sign., vol. 3, (2002), 84–91
- [4] Isobe S., Zhou X., Nishizeki T.: *Cost total colorings of trees*. IEICE Trans. INF. & SYST., vol. E-87, (2004), 337–342
- [5] Kao M., Lam T., Sung W., Ting H.: *All-cavity maximum matchings*. Proc. ISAAC'97, LNCS, 1350, (1997), 364–373
- [6] Krawczyk H., Kubale M.: *An approximation algorithm for diagnostic test scheduling in multi-computer systems*. IEEE Trans. Comput., C-34, (1985), 869–872
- [7] Marx D.: *The complexity of tree multicolorings*. Proc. MFCS'02, LNCS, 2420, (2002), 532–542
- [8] Marx D.: *List edge multicoloring in graphs with few cycles*. Inf. Proc. Lett., 89, (2004), 85–90
- [9] Micali S., Vazirani V.: *An $O(mn^{1/2})$ algorithm for finding maximum matching in general graphs*. Proc. 21st Ann. IEEE Symp. on Foundations of Computer Science, (1980), 17–27
- [10] Zhou X., Nishizeki T.: *Algorithms for the cost edge-coloring of trees*. LNCS, 2108, (2001), 288–297