ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1992

Seria: AUTOMATYKA z. 110 Nr kol. 1176

Marek Kubale
Politechnika Gdańska

SZEREGOWANIE TRANSMISJI PLIKÓW W OKNACH CZASOWYCH

FILE TRANSFER SCHEDULING WITHIN TIME WINDOWS

HARMONOGRAMMBILDUNG FÜR DATEIENÜBERTRAGUNG IN ZEITFENSTERN

Streszczenie: Niniejsza praca poświęcona jest problemowi sterowania przepływem plików danych w sieciach komputerowych z punktu widzenia sprawności procesu transmisji. Pokazano, że problem szeregowania plików w oknach czasowych jest NP-trudny nawet wówczas, gdy sieć ma strukturę gwiaździstą. Następnie wprowadzono dolne i górne oszacowanie długości optymalnego uszeregowania w przypadku ogólnym. W ostatniej części podano wielomianowe algorytmy konstruowania optymalnego harmonogramu dla przypadków szczególnych sieci pętlowych i gwiaździstych.

Summary: This paper is devoted to the problem of scheduling file transfers in computer networks from the complexity point of view. It is shown that the problem of scheduling files within time windows is NP-hard even if the network has a star topology. Next, lower and upper bounds on the optimal makespan are derived. Finally, polynomial-time algorithms for constructing optimal schedules in some special cases of star and loop networks are given.

Zusammenfassung: Im vorliegenden Beitrag ist die Leistung von der Flussteuerung der Dateien in Rechnernetzen erwogen. Gezeigt ist, daß die Harmonogrammbildung für Dateienübertragung in Zeitfenstern, auch in Netzen mit Sternstruktur, NP-schwach ist. Danach ist die untere und obere Bewertung der Länge des optimalen Harmonogramms in allgemeinen Fall abgeleitet. Einige Polynomial-Algorithmen, zum Bau optimaler Harmonogramme für bestimmte Schleife- und Sternnetze, sind am Ende vorgeschlagen.

1. Wstęp

Jednym z podstawowych problemów występujących w trakcie działania sieci komputerowych jest zagadnienie przesyłania dużych plików danych pomiędzy serwerami i terminalami. Niniejsza praca poświęcona jest problemowi sterowania przepływem takich plików z punktu widzenia sprawności procesu transmisji. Przyjmujemy tutaj, że program wykonania wszystkich transmisji między węzłami sieci dany jest z góry. Zakładamy również, że przesłania mogą odbywać się jedynie w takich przedziałach czasu, w których oba komputery: nadający i odbierający są wolne, tzn. nie są zaangażowane w inne transmisje bądź inne prace obliczeniowe. Celem naszym jest znalezienie najkrótszego harmonogramu wykonania transmisji bez przerwań.

Modelem matematycznym takiego szeregowania zadań jest graf obciążony $G = (V,E)$ o zbiorze wierzchołków V odpowiadających komputerom sieci i zbiorze krawędzi E oznaczających przewidywane transmisje plikowe. Wagą krawędzi (u,v) jest rozmiar pliku, który ma być przesłany między węzłem u i v. Oczywiście rozmiar pliku może być również traktowany jako czas trwania odpowiedniej transmisji. Ponadto, z każdym wierzchołkiem v związana jest

..kończona liczba (być może 0) przedziałów zajętości, czyli niedostępności odpowiedniego komputera do transmisji.

W niniejszej pracy pokażemy, że problem szeregowania transmisji plików jest NP-trudny nawet wówczas, gdy sieć ma strukturę gwiaździstą. Następnie wprowadzimy dolne i górne oszacowanie długości optymalnego uszeregowania w przypadku ogólnym. W ostatniej części podamy wielomianowe algorytmy budowania optymalnego harmonogramu dla przypadków szczególnych sieci pętlowych i gwiaździstych.

2. Dowód NP-zupełności

Aby przedstawić bardziej formalnie cel naszych rozważań, wprowadzimy następujący model matematyczny. Niech będzie dany zbiór m plików $P = \{p_1,p_2,\ldots,p_m\}$, które mają być przesłane pomiędzy n maszynami $M = \{m_1,m_2,\ldots,m_n\}$. Każdy plik p_i przesyłany jest pomiędzy dwiema maszynami $m_j,m_k \in M$ przez okres równy $t_i = t(p_i)$ jednostek czasu. Zakładamy, że momenty rozpoczynania i kończenia wszystkich zadań w systemie są wielokrotnościami podstawowej jednostki czasu. Ponadto, z każdą maszyną m_i związana jest skończona liczba przedziałów zajętości, czyli niedostępności danego komputera do transmisji. Oznacza to, iż każda maszyna ma tyle samo lub o 1 więcej przedziałów, w których może przyjmować lub wysyłać pliki. Tak więc z i-tą maszyną związana jest sekwencja momentów czasu $O_i = (r_1,d_1,r_2,d_2,\ldots)$, gdzie r_j jest początkiem, a d_j końcem j-tego przedziału dostępności do transmisji. Sekwencję O_i nazywać będziemy *oknem czasowym* związanym z maszyną m_i.

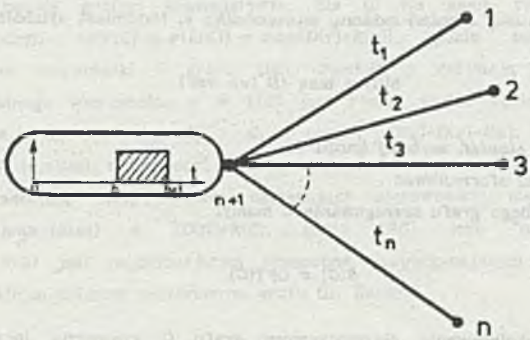
Rozważany problem możemy przedstawić w postaci grafu nieskierowanego $G = (V,E)$ z obciążonymi wierzchołkami i krawędziami. Zbiór wierzchołków V odpowiada zbiorowi maszyn M, zaś zbiór krawędzi E odpowiada zbiorowi plików P. Każda krawędź $e_i = \{v_j,v_k\}$ ma wagę $t_i = t(p_i)$. Ponadto, każdy wierzchołek v_i ma okno O_i gotowości do transmisji. Taki graf G nazywać będziemy *grafem szeregowania*.

Łatwo zauważyć, że rozwiązaniem naszego problemu szeregowania transmisji plików w oknach czasowych jest pokolorowanie przedziałowe krawędzi grafu szeregowania w obecności kolorów zabronionych. Jest tak dlatego, iż każdy kolor przydzielony dowolnej krawędzi można potraktować jako jednostkę czasu, w której wykonuje się odpowiednia transmisja. Tak więc nasz problem szeregowania uogólnia problem kolorowania krawędzi, bowiem ten drugi sprowadza się po prostu do znalezienia rozwiązania w przypadku, gdy wszystkie pliki mają długość 1 i nie ma żadnych okien czasowych. Z drugiej strony wiadomo, że problem kolorowania krawędzi należy do klasy problemów NP-trudnych. Ściślej, problem określenia indeksu chromatycznego (a więc i klasy grafu prostego) jest NP-zupełny [4]. W praktyce oznacza to, że oba problemy nie mogą być rozwiązane w czasie wielomianowym.

Obecnie pokażemy, że problem szeregowania transmisji plikowych jest NP-trudny nawet wówczas, gdy sieć komputerowa ma strukturę gwiaździstą. W tym celu przez SZEREGOWANIE TRANSMISJI PLIKÓW w OKNACH CZASOWYCH (STPOC) oznaczymy wersję decyzyjną rozważanego problemu.

Twierdzenie 1. STPOC jest NP-zupełny nawet wówczas, gdy graf szeregowania jest gwiazdą i brak ograniczeń czasowych we wszystkich wierzchołkach z wyjątkiem centralnego.

Dowód. Przynależność STPOC do klasy NP jest oczywista, dlatego skoncentrujemy się na redukcji. Jako znany problem NP-zupełny wybieramy PODZIAŁ zbioru, który zdefiniowany jest następująco: "Dany jest zbiór liczb naturalnych $A = \{a_1, a_2, \ldots, a_n\}$ taki, że $\sum a_l = 2b$, gdzie $l \in \{1,\ldots,n\} = Q$. Czy istnieje podzbiór $P \subset Q$, dla którego $\sum_{l \in P} a_l = \sum_{l \notin P} a_l = b$?" Fakt, że PODZIAŁ zbioru jest problemem NP-zupełnym został udowodniony w [5].



Rys. 1. Graf szeregowania $K_{1,n}$

Fig. 1. Scheduling graph $K_{1,n}$

Mając dany zbiór A budujemy graf $K_{1,n}$ i obciążamy jego krawędzie wagami $a_1, a_2, \ldots, a_n$. Następnie przyjmujemy $O_l = (0,\infty)$ dla każdego $l = 1,2,\ldots,n$ oraz $O_{n+1} = (0, b, b+1, \infty)$ dla wierzchołka centralnego v_{n+1}. Przykład takiej gwiazdy pokazany jest na rys. 1.

Łatwo zauważyć, że odpowiedni podzbiór P dla liczb A istnieje wtedy i tylko wtedy, gdy istnieje uszeregowanie transmisji plików o długości nie przekraczającej $2b+1$. Problem PODZIAŁ-u zbioru może być zatem rozwiązany dzięki rozwiązaniu odpowiedniego przypadku problemu szeregowania transmisji plików w sieci gwiaździstej. Dlatego problem STPOC jest NP-zupełny. □

3. Oszacowania długości uszeregowań optymalnych

W poprzednim punkcie pokazaliśmy, że problem szeregowania transmisji plików jest NP-trudny w przypadku ogólnym i wielu przypadkach szczególnych. W praktyce oznacza to, że nie istnieją algorytmy budowania harmonogramów optymalnych, które działają w czasie wielomianowym. Jednakże, w czasie ograniczonym wielomianem można obliczyć dolne i górne oszacowanie długości odpowiednich harmonogramów optymalnych.

Długość optymalnego uszeregowania związanego z grafem G oznaczymy przez OPT(G). Ponadto przez $r(l)$ oznaczymy początek pierwszego przedziału wolnego do transmisji pliku p_l, przez $R(l)$ zaś oznaczymy początek ostatniego przedziału dostępności. Niech $E(v)$ będzie zbiorem wszystkich krawędzi incydentnych z $v \in V$. Wówczas przez *stopień ważony wierzchołka* v rozumieć będziemy:

$$D(v) = \sum_{e \in E(v)} t(e), \tag{1}$$

gdzie $t(e)$ jest wagą odpowiedniej krawędzi. Następnie przez

$$D^*(v) = D(v) + \min\{r(e): e \in E(v)\} \tag{2}$$

oznaczymy tzw. *uzupełniony stopień ważony wierzchołka* v. Natomiast symbolem

$$\delta(G) = \max\{D^*(v): v \in V\} \tag{3}$$

oznaczymy *uzupełniony stopień ważony grafu* G.

Obecnie możemy już sformułować

Twierdzenie 2. Dla każdego grafu szeregowania G mamy:

$$\delta(G) \leq \mathrm{OPT}(G). \tag{4}$$

Dowód. W każdym pokolorowaniu chromatycznym grafu G krawędzie incydentne z dowolnym wierzchołkiem v muszą otrzymać rozłączne przedziały kolorów, przy czym najniższy kolor przydzielony krawędzi musi być większy od $\min\{t(e)\}$, gdzie minimum jest rozciągnięte na wszystkie krawędzie $e \in E(v)$. Wybierając "najcięższy" taki stopień otrzymujemy żądane oszacowanie dolne. ☐

Rzecz jasna, oszacowanie (4) może być bardzo nieprecyzyjne. W szczególności może zdarzyć się, że w grafie G istnieje krawędź $e = (u,v)$ taka, że $t(e)+r(e) \gg \max\{D^*(u), D^*(v)\}$ a nawet, że $t(e)+r(e) \gg \delta(G)$. Prowadzi to nas do kolejnego dolnego oszacowania postaci:

$$\max\{t(e)+r(e): e \in E\} \leq \mathrm{OPT}(G). \tag{5}$$

Oba oszacowania (4) i (5) można otrzymać w czasie $O(m+n)$.

Obecnie, w celu wyprowadzenia efektywnego oszacowania wartości $\mathrm{OPT}(G)$ z góry, zdefiniujemy pojęcie *sąsiedztwa ważonego krawędzi* $e \in E$ jako:

$$N(e) = D(u)+D(v)-t(e), \tag{6}$$

gdzie $e = (u,v)$. Zatem $N(e)$ jest wagą wszystkich krawędzi sąsiadujących z e plus waga samej krawędzi e. Przez *uzupełnione sąsiedztwo ważone krawędzi* e rozumiemy wartość:

$$N^*(e) = N(e)+R(e). \tag{7}$$

Natomiast następujący niezmiennik grafu G:

$$\sigma(G) = \max\{N^*(e): e \in E\} \tag{8}$$

określimy jako *uzupełnione sąsiedztwo ważone grafu G*.

Twierdzenie 3. Dla każdego grafu szeregowania G mamy:

$$\mathrm{OPT}(G) \leq \sigma(G). \tag{9}$$

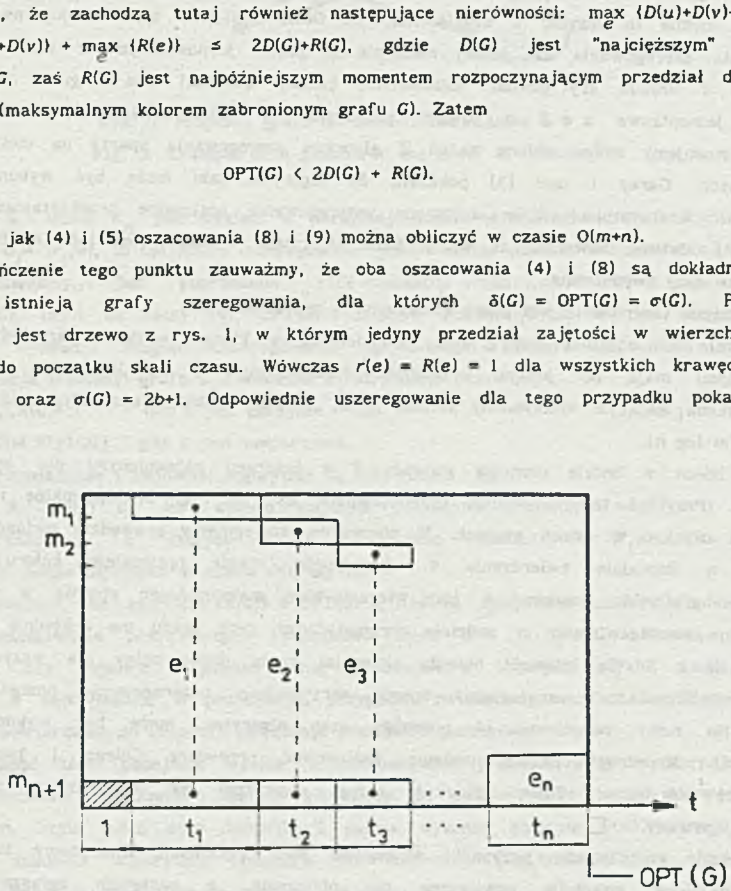
Dowód. Niech $L(G)$ będzie grafem krawędziowym dla G. Na mocy twierdzenia 2 w [6], $\mathrm{OPT}(L(G)) \leq \sigma(L(G))$, czyli $\mathrm{OPT}(G) \leq \sigma(L(G)) = \max\{D(v)+R(v)\}$, gdzie maksimum jest rozciągnięte na wszystkie wierzchołki v grafu $L(G)$. Zgodnie z definicją grafu krawędziowego waga sąsiedztwa dowolnego wierzchołka v w $L(G)$ jest równa wadze sąsiedztwa odpowiadającej mu krawędzi $e = \{u,v\}$ w grafie G, czyli $D(u)+D(v)-t(e)$. Zatem $\sigma(L(G)) = \max\{D(u)+D(v)+R(e)-t(e)\colon \{u,v\}=e\}$, co dowodzi prawdziwości oszacowania (9). □

Zauważmy, że zachodzą tutaj również następujące nierówności: $\max\limits_{e} \{D(u)+D(v)+R(e)-t(e)\} < \max\limits_{u,v} \{D(u)+D(v)\} + \max\limits_{e} \{R(e)\} \leq 2D(G)+R(G)$, gdzie $D(G)$ jest "najcięższym" stopniem ważonym w G, zaś $R(G)$ jest najpóźniejszym momentem rozpoczynającym przedział dostępności do transmisji (maksymalnym kolorem zabronionym grafu G). Zatem

$$\mathrm{OPT}(G) < 2D(G) + R(G). \tag{10}$$

Podobnie jak (4) i (5) oszacowania (8) i (9) można obliczyć w czasie $O(m+n)$.

Na zakończenie tego punktu zauważmy, że oba oszacowania (4) i (8) są dokładne w tym sensie, że istnieją grafy szeregowania, dla których $\delta(G) = \mathrm{OPT}(G) = \sigma(G)$. Przykładem takiego grafu jest drzewo z rys. 1, w którym jedyny przedział zajętości w wierzchołku v_{n+1} sprowadzono do początku skali czasu. Wówczas $r(e) = R(e) = 1$ dla wszystkich krawędzi $e \in E$ i $\delta(G) = 2b+1$ oraz $\sigma(G) = 2b+1$. Odpowiednie uszeregowanie dla tego przypadku pokazane jest na rys. 2.



Rys. 2. Wykres Gantta dla grafu gwiaździstego

Fig. 2. Grantt diagram for a star graph

4. Przypadki rozwiązywalne wielomianowo

W punkcie 2 pokazaliśmy, że problem szeregowania transmisji plików jest NP-trudny nawet wówczas, gdy graf szeregowania jest gwiazdą. Dlatego jest dosyć trudno podać nietrywialne przypadki szczególne, które mogą być rozwiązane za pomocą algorytmu wielomianowego. Tym niemniej, w dalszej części tego punktu opiszemy kilka takich sytuacji. Będą one dotyczyły na ogół grafów rzadkich, w których liczba krawędzi nie przewyższa liczby wierzchołków. Ponadto ograniczymy się do przypadków, w których krawędzie mają jednakową wagę równą l jednostkom czasu.

Twierdzenie 4. Jeśli każda krawędź gwiazdy $K_{1,n}$ ma wagę l i dokładnie jeden przedział dostępności do transmisji, to uszeregowanie optymalne można otrzymać w czsie $O(n \cdot \log n)$.

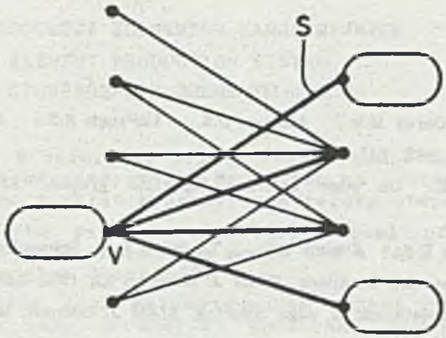
Dowód. Najpierw obliczamy górne oszacowanie $\sigma(G)$ długości uszeregowania optymalnego. Jak wiadomo, można to uczynić w czasie liniowym $O(n)$. Następnie sprowadzamy nasz problem do zagadnienia szeregowania na jednej maszynie n zadań jednostkowych z ułamkowymi czasami przybycia i liniami krytycznymi. Mianowicie, każdej krawędzi $e \in E$ grafu $K_{1,n}$ odpowiada zadanie jednostkowe $z \in Z$ z czasem przybycia $r_z = r(e)/l$ i linią krytyczną $d_z = d(e)/l$. Obecnie stosujemy wobec zbioru zadań Z algorytm szeregowania oparty na metodzie regionów zabronionych. Garey i inni [3] pokazali, że algorytm taki może być wykonany w czasie $O(n \cdot \log n)$. W ostatnim etapie otrzymane uszeregowanie optymalne przekształcamy do postaci wyjściowej. Łatwo zauważyć, że każda faza algorytmu wymaga $O(n)$ lub $O(n \cdot \log n)$ operacji, co dowodzi tezy twierdzenia. □

Następne twierdzenie jest prostą konsekwencją poprzedniego.

Twierdzenie 5. Jeżeli G jest grafem dwudzielnym, w którym wszystkie krawędzie z kolorami zabronionymi mają po jednym przedziale dopuszczalnym i są incydentne z tym samym wierzchołkiem, mającym maksymalny stopień w G, to uszeregowanie optymalne można otrzymać w czasie $O(m \cdot \log n)$.

Dowód. Niech v będzie centrum gwiazdy S z kolorami zabronionymi dla jej i tylko jej krawędzi. Przykład takiej sytuacji przedstawiony jest na rys. 3. Optymalne rozwiązanie dla G można uzyskać w dwóch etapach. W pierwszym szeregujemy krawędzie gwiazdy S w sposób opisany w dowodzie twierdzenia 4. W drugim etapie optymalnie kolorujemy wszystkie krawędzie grafu G. Ponieważ v jest wierzchołkiem maksymalnego stopnia w G, więc każde skojarzenie tworzące kolor w podziale chromatycznym tego grafu ma dokładnie jedną krawędź pochodzącą z S. Ta krawędź określa przedział czasu dopuszczalny dla wszystkich krawędzi owego skojarzenia. W analogiczny sposób otrzymujemy uszeregowanie pozostałych krawędzi grafu. Na mocy twierdzenia 4 pierwszy etap algorytmu może być wykonany w czasie $O(n \cdot \log n)$. W drugiej fazie możemy zastosować procedurę Colego i Hopcrofta [2] o złożoności $O(m \cdot \log n)$. Zatem złożoność całego algorytmu jest zdominowana przez złożoność drugiej jego fazy. □

Obecnie rozpatrzymy przypadki szczególne sieci pętlowych. Zakładamy tutaj, że każdy węzeł najpierw wykonuje powierzone mu obliczenia, a następnie zgłasza gotowość do transmisji plików.



Rys. 3. Przykład grafu do dowodu twierdzenia 5

Fig. 3. Example of a graph for the proof of Theorem 5

Twierdzenie 6. Jeżeli C_n jest cyklem o parzystej liczbie wierzchołków, z których każdy ma okno postaci $O_i = (r_i, \infty)$, to optymalne uszeregowanie plików można uzyskać w czasie $O(n)$.

Dowód. Oznaczmy, jak poprzednio, $R(G) = \max\{r_i:\ i=1,2,\ldots,n\}$. Łatwo zauważyć, że $\delta(G) = 2l+R(G)$, czyli na mocy (4), $OPT(G) \geq 2l+R(G)$. Pokażemy, że w tym przypadku dolne oszacowanie wyznacza długość uszeregowania optymalnego. Rzeczywiście, niech będzie dana ciągła numeracja krawędzi grafu C_n. Wówczas e_i otrzymuje interwał:

$[R(G), R(G)+l]$ gdy i jest parzyste lub

$[R(G)+l, R(g)+2l]$ gdy i jest nieparzyste.

Poprawność rozwiązania i liniowość algorytmu są oczywiste. □

Twierdzenie 7. Jeżeli C_n jest cyklem mającym jednakowe krawędzie wagi l i nieparzystą liczbę wierzchołków, z których każdy ma okno postaci $O_i = (r_i, \infty)$, to optymalne uszeregowanie można otrzymać w czasie $O(n \cdot \log l)$.

Dowód. Zauważmy, że $\max\{3l, 2l+R(G)\} \leq OPT(G) \leq 3l+R(G)$. Optymalne uszeregowanie otrzymamy metodą przeszukiwania połówkowego. Początkowo niech $k = \lfloor 5l/2 \rfloor + R(G)$. Metodą sekwencyjną sprawdzamy, czy istnieje k-pokolorowanie przedziałowe krawędzi cyklu. Jeśli tak, to zmniejszamy k odpowiednio, w przeciwnym przypadku zwiększamy wartość k. Etap weryfikacji rozwiązania powtarzamy do momentu zawężenia przedziału poszukiwań do 1.

Poprawność tego podejścia wynika z zastosowania przeszukiwania wyczerpującego. Aby oszacować złożoność obliczeniową metody zauważmy, że pokolorowanie każdej krawędzi wymaga stałego czasu, czyli jeden etap weryfikacji można wykonać kosztem $O(n)$ operacji. Ponieważ w trakcie działania algorytmu wykonuje się najwyżej $\lceil \log_2 \min\{l, R(G)\} \rceil \leq \lceil \log_2 l \rceil$ takich kroków dla różnych k, więc algorytm ten wymaga czasu $O(n \cdot \log l)$. □

Na zakończenie tej pracy odnotujmy dwie inne publikacje, ściśle wiążące się z jej tematyką. Pierwszą jest doskonały artykuł Coffmana i innych [1], poświęcony szeregowaniu transmisji plików bez ograniczeń czasowych. Druga praca [7] dotyczy przedziałowego kolorowania krawędzi grafu w obecności kolorów zabronionych, które jest naturalnym modelem matematycznym dla zagadnień rozważanych powyżej.

LITERATURA

[1] Coffman E.G., Jr., Garey M.R., Johnson D.S., LaPaugh A.S.: Scheduling file transfers, *SIAM J. Comput.* 14, 1985, 744-780.

[2] Cole R., Hopcroft J.E.: On edge coloring bipartite graphs, *SIAM J. Comput.* 11, 1982, 540-546.

[3] Garey M.R., Johnson D.S., Simons B.B., Tarjan R.E.: Scheduling unit-time tasks with arbitrary release times and deadlines, *SIAM J. Comput.* 10, 1981, 256-269.

[4] Holyer I.: The NP-completeness of edge coloring, *SIAM J. Comput.* 10, 1981, 718-720.

[5] Karp R.M.: Reducibility among combinatorial problems, w: *Complexity of Computer Computations* (R.E. Miller, J.W. Thather, red.), Plenum Press, New York, 1972, 85-103.

[6] Kubale M.: Interval vertex-coloring of a graph with forbidden colors, *Disc. Math.* 74, 1989, 125-136.

[7] Kubale M.: Interval edge-coloring of a graph with forbidden colors, *Disc. Math.* (to appear).

Recenzent: Prof.dr h.inż. Jan Węglarz

Wpłynęło do Redakcji do 30.04.1992 r.

Abstract:

One of the basic problems occurring in the operation of computer networks is transferring large files between various terminals and servers. This paper is devoted to the problem of scheduling such transfers from the complexity point of view. We are interested in how collections of such file transfers can be scheduled so as to minimize the total time for the overall transfer process under the restriction that certain nodes cannot take part in it within prespecified time intervals, i.e., when either of nodes: sending and receiving is angaged in other important computation. In this paper we show that the scheduling problem is NP-hard even if the network has a star topology. Next, polynomial lower and upper bounds on the optimal makespan are derived. Finally, polynomial-time algorithms for costructing optimal schedules in some special cases of star and loop networks are given.