ZESZYTY NAUKOWE POLITECHNIKI BIAŁOSTOCKIEJ 2008
Informatyka – Zeszyt 3

Leon Bobrowski[1,2], Tomasz Łukaszuk[1]

SZEREGOWANIE ZADAŃ OBLICZENIOWYCH Z ZASTOSOWANIEM MODELU RANGOWEGO

Streszczenie: Zagadnienia szeregowania zadań pojawiają się między innymi w kontekście problemów realizowalności dużych procesów obliczeniowych i ich optymalizacji. Przy rozstrzyganiu tego typu problemów można wykorzystywać metody regresji rangowej. Do celów konstrukcji modeli regresji rangowej poszczególne zadania obliczeniowe charakteryzowane są poprzez wielowymiarowe wektory zależności. Wektory zależności pozwalają stwierdzić czy określone zadanie może być zrealizowane tylko wtedy, gdy zostaną wcześniej zrealizowane pewne inne zadania. Regresja rangowa obejmuje konstrukcję takich odwzorowań liniowych z wielowymiarowej przestrzeni zależności na przestrzeń jednowymiarową (linię czasu), która odzwierciedla w możliwie dużym stopniu zależności pomiędzy zadaniami.

Słowa kluczowe: szeregowanie zadań obliczeniowych, model rangowy, wypukła i odcinkowo-liniowa funkcja kryterialna (CPL)

1. Wstęp

Realizację dużych zadań obliczeniowych wiąże się na ogół z dążeniem do maksymalnego skracania czasu obliczeń przy spełnieniu wszystkich warunków ograniczających [4], [7], [6]. Dla skrócenia czasu obliczeń, procesy obliczeniowe dekomponuje się na podzadania O_j, które następnie realizuje się szeregowo poprzez jeden procesor lub w sposób równoległy przy o większej liczbie procesorów. Modele szeregowania zadań pomocne są zarówno przy weryfikacji warunków realizowalności zdekomponowanych programów, jak również przy skracaniu czasu obliczeń poprzez odpowiednie rozdzielanie zadań na poszczególne procesory. W przedstawionym artykule analizowana jest metoda badania niesprzeczności, szeregowania oraz optymalizacji zadań obliczeniowych O_j, oparta na koncepcjach regresji rangowej [2].

Modele regresji rangowej mają postać takich odwzorowań liniowych z wielowymiarowej przestrzeni cech na linię prostą, które w możliwie dużym stopniu zachowują zadany porządek w wybranych parach zdarzeń lub obiektów [2]. Porządek

[1] Wydział Informatyki, Politechnika Białostocka, Białystok
[2] Instytut Biocybernetyki i Inżynierii Biomedycznej, PAN, Warszawa

realizacji dwu zadań O_j i O_k oznacza, że do realizacji zadania O_k potrzebna jest znajomość pewnych wyników realizacji zadania O_j. W rezultacie wcześniejsza realizacja zadania O_j jest warunkiem koniecznym realizacji zadania O_k. Wymagany porządek realizacji zadań obliczeniowych można reprezentować w postaci wielowymiarowych wektorów zależności. Wektor zależności zadania O_k identyfikuje takie zadania O_j $(j \neq k)$, których wcześniejsza realizacja jest warunkiem koniecznym realizacji zadania O_k [1].

Konstrukcja rangowych odwzorowań liniowych została oparta na wyznaczaniu hiperłaszczyzny możliwie najlepiej rozdzielającej dwa zbiory w przestrzeni zależności. Wyznaczanie hiperłaszczyzny rozdzielającej można oprzeć na minimalizacji wypukłej i odcinkowo liniowej (*CPL*) funkcji kryterialnej [2]. Funkcja kryterialna definiowana jest w tym przypadku na podstawie wektorów zależności charakteryzujących poszczególne zadania obliczeniowe O_k. Algorytmy wymiany rozwiązań bazowych, podobne do programowania liniowego, pozwalają wyznaczać minimum funkcji typu *CPL* w sposób efektywny nawet w przypadku dużej liczby wektorów zależności.

2. Charakterystyka zadań obliczeniowych przez wektory zależności

Załóżmy, że realizowany proces obliczeniowy został podzielony (zdekomponowany) na m zadań O_k $(k = 1, ..., m)$. Każde z zadań O_k zostało scharakteryzowane przez czas realizacji τ_k oraz przez wektor zależności $\rho_\mathbf{k} = [\rho_{k1}, ..., \rho_{km}]^T$ o m składowych binarnych ρ_{kj} ($\rho_{kj} = 1$ lub $\rho_{kj} = 0$). Wartość $\rho_{kj} = 1$ oznacza, że k-te zadanie O_k może być realizowane tylko wtedy, gdy wcześniej będzie zrealizowane zadanie O_j. Wartość $\rho_{kj} = 0$ oznacza, że k-te zadanie O_k może być realizowane niezależnie od stanu realizacji zadania O_j. Zakładamy przy tym, że zadania O_k i O_j nie mogą blokować się wzajemnie, tj.:

$$(\forall k, j \in \{1, ..., m\})\ \rho_{kk} = 0 \quad \mathit{oraz} \quad (\rho_{kj} = 1) \Rightarrow (\rho_{jk} = 0) \tag{1}$$

Na podstawie wektorów zależności $\rho_\mathbf{k}$ wprowadzimy rangową *relację poprzedzania* $O_j \prec O_k$ pomiędzy zadaniami O_j i O_k, która oznacza, że zadanie O_j poprzedza O_k:

$$(O_j \prec O_k) \Leftrightarrow (\rho_{kj} = 1) \tag{2}$$

Przyjmujemy, że zarówno podział procesu obliczeniowego na zadania O_k jak również charakterystyka tych zadań za pomocą wektorów zależności ρ_k oraz czasów obliczeń τ_k wynikają z wiedzy ekspertów o realizowanym procesie. Np. w hierarchicznej strukturze warstwowej realizacja każdego zadania $O_{k(l)}$ w l-tej warstwie jest możliwa tylko wtedy, gdy wszystkie zadania $O_{k(l-1)}$ w warstwie wcześniejszej zastaną zrealizowane. Przy realizacji złożonych procesów obliczeniowych liczba m zadań O_k może być bardzo duża. W takich okolicznościach pojawia się zagrożenie, że proponowany przez ekspertów podział procesu na zadania O_k, scharakteryzowane za pomocą wektorów zależności $\rho_\mathbf{k}$ oraz czasów obliczeń τ_k, spowoduje pojawienie się trudności w realizacji procesu obliczeniowego. Realizacja procesu może być niemożliwa z powodu pojawienia się ciągu zależności sprzecznych, wzajemnie blokujących się. Blokowanie się procesu obliczeniowego może nastąpić na przykład wówczas, gdy zadanie O_k odwołuje się do zadania O_l, a zadanie O_l odwołuje się do zadania O_m, które może być zrealizowane po wykonaniu zadania O_k. W tym przypadku podzbiór rangowych relacji poprzedzania (2) ma poniższą postać:

$$\{(O_l \prec O_k), (O_m \prec O_l), (O_k \prec O_m)\} \tag{3}$$

Relacja $\prec$ jest przechodnia (ang. *transient relation*) wtedy i tylko wtedy, gdy dla dowolnych O_k, O_l, O_m spełniona jest poniższa implikacja:

$$jezeli \quad (O_l \prec O_k) \quad oraz \quad (O_k \prec O_m), \quad to \quad (O_m \prec O_l) \tag{4}$$

Zauważmy, że pary ze zbioru (3) nie tworzą relacji przechodniej. Reprezentacja zbioru (3) za pomocą grafu skierowanego z wierzchołkami utworzonymi przez zadania O_k pozwala uwypuklić istnienie pętli, która jest źródłem braku przechodniości i zablokowania się procesu.

Blokujące się łańcuchy zależności mogą mieć oczywiście długość większą niż trzy. Łatwo weryfikowalne warunki (1) zabezpieczają przed blokowaniem się wzajemnym w parach zadań. Weryfikacja zagrożenia sprzecznościami i blokowaniem się w długich sekwencjach zadania O_k wymaga wprowadzenia innych sposobów analizy. Istnieje możliwość wykorzystania do tego celu regresji rangowej [2]. Innym problemem, który może być również analizowany za pomocą regresji rangowej jest rozpatrywanie możliwości skrócenia czasu realizacji procesu obliczeniowego, poprzez równoległą realizację pewnych zadań O_k. Model regresji rangowej pozwala w pewnych przypadkach wskazać takie zadania O_k, których równoległa realizacja skróci czas przebiegu procesu.

3. Liniowe odwzorowania rangowe

Rozważmy liniowe odwzorowanie $t(\mathbf{w}) = \mathbf{w}^T\rho$, gdzie $\mathbf{w} = [w_1, ..., w_m]^T$ $(\mathbf{w} \in R^m)$ jest wektorem parametrów, a $\rho = [\rho_\mathbf{1}, ..., \rho_\mathbf{m}]$ macierzą utworzoną z m-wymiarowych wektorów zależności $\rho_\mathbf{k}$ (1). Prosta $t(\mathbf{w}) = \mathbf{w}^T\rho$ może reprezentować kolejność rozpoczęcia realizacji t_k poszczególnych zadań O_k:

$$(\forall k \in \{1,...,m\}) \quad t_k = \mathbf{w}^T\rho_\mathbf{k} \tag{5}$$

Projektując rangowy model zależności poszukujemy takiego optymalnego wektora parametrów $\mathbf{w}^*$, dla którego odwzorowanie liniowe $t(\mathbf{w}) = \mathbf{w}^T\rho$ w najlepszym stopniu zachowuje poniższe *implikacje rangowe*:

$$\begin{array}{l} O_j \prec O_k \Rightarrow (\mathbf{w}^*)^T\rho_\mathbf{j} < (\mathbf{w}^*)^T\rho_\mathbf{k} \quad oraz \\ O_k \prec O_j \Rightarrow (\mathbf{w}^*)^T\rho_\mathbf{j} > (\mathbf{w}^*)^T\rho_\mathbf{k}, \quad gdzie\ j < k \end{array} \tag{6}$$

Metodę poszukiwania optymalnego wektora parametrów $\mathbf{w}^*$ (6) można oprzeć na badaniu *liniowej rozdzielności* za pomocą hiperpłaszczyzny przechodzącej przez początek układu współrzędnych przestrzeni cech poniższych dwu zbiorów *wektorów różnicowych* $\mathbf{r_{jk}} = \rho_\mathbf{k} - \rho_\mathbf{j}$, gdzie $j < k$:

$$\begin{array}{l} R^+ = \{\mathbf{r}_{jk} = \rho_\mathbf{k} - \rho_\mathbf{j} : \ O_j \prec O_k\} \\ R^- = \{\mathbf{r}_{jk} = \rho_\mathbf{k} - \rho_\mathbf{j} : \ O_k \prec O_j\} \end{array} \tag{7}$$

Definicja 1. *Zbiory R^+ i R^- są liniowo separowalne za pomocą hiperpłaszczyzny przechodzącej przez początek układu współrzędnych wtedy i tylko wtedy, gdy istnieje taki wektor parametrów $\mathbf{w}'$, że zachodzą poniższe nierówności:*

$$\begin{array}{r} (\forall \mathbf{r}_{jk} \in R^+)\ \mathbf{w}'\mathbf{r}_{jk} > 0 \\ oraz\ (\forall \mathbf{r}_{jk} \in R^-)\ \mathbf{w}'\mathbf{r}_{jk} < 0 \end{array} \tag{8}$$

Wektor parametrów $\mathbf{w}'$ wyznacza hiperpłaszczyznę $H(\mathbf{w}')$ w przestrzeni cech (przestrzeni zależności) przechodzącą przez początek układu współrzędnych tej przestrzeni:

$$H(\mathbf{w}') = \{\mathbf{x} : \ \mathbf{w}'\mathbf{x} = 0,\ \mathbf{x} \in R^m\} \tag{9}$$

O hiperpłaszczyznie $H(\mathbf{w}')$ mówimy, że separuje zbiory R^+ i R^- (7). Wszystkie elementy $\mathbf{r}_{jk}$ zbioru R^+ leżą po dodatniej stronie hiperpłaszczyzny $H(\mathbf{w}')$, a wszystkie elementy $\mathbf{r}_{jk}$ zbioru R^- leżą po jej stronie ujemnej.

Można wykazać słuszność poniższego Lematu [2]:

Lemat 1. *Odwzorowanie liniowe $t = t(\mathbf{w}') = (\mathbf{w}')^T\rho$ zachowuje wszystkie implikacje (6), wtedy i tylko wtedy, gdy hiperpłaszczyzna $H(\mathbf{w}')$ (9) wyznaczona przez wektor $\mathbf{w}'$ separuje zbiory R^+ i R^- (7).*

4. Wypukła i odcinkowo liniowa (*CPL*) funkcja kryterialna $\Phi(\mathbf{w})$

Projektowanie hiperpłaszczyzny $H(\mathbf{w}^*)$ (9) optymalnie rozdzielającej zbiór dodatni R^+ od zbioru ujemnego R^- (7) może być oparte na minimalizacji wypukłej i odcinkowo-liniowej (*CPL*) funkcji kryterialnej $\Phi(\mathbf{w})$, która jest podobna do perceptronowej funkcji kryterialnej [6]. Wprowadźmy w tym celu pozytywne funkcje kary $\varphi^+_{jk}(\mathbf{w})$ określone na elementach $\mathbf{r}_{jk}$ zbioru R^+ oraz negatywne funkcje kary $\varphi^-_{jk}(\mathbf{w})$ określone na elementach $\mathbf{r}_{jk}$ zbioru R^- (7):

$$(\forall \mathbf{r}_{jk} \in R^+) \quad \varphi^+_{jk}(\mathbf{w}) = \begin{cases} 1 - \mathbf{w}^T\mathbf{r}_{jk} & jezeli \quad \mathbf{w}^T\mathbf{r}_{jk} \le 1 \\ 0 & jezeli \quad \mathbf{w}^T\mathbf{r}_{jk} > 1 \end{cases} \tag{10}$$

oraz

$$(\forall \mathbf{r}_{jk} \in R^-) \quad \varphi^-_{jk}(\mathbf{w}) = \begin{cases} 1 + \mathbf{w}^T\mathbf{r}_{jk} & jezeli \quad \mathbf{w}^T\mathbf{r}_{jk} \ge -1 \\ 0 & jezeli \quad \mathbf{w}^T\mathbf{r}_{jk} < -1 \end{cases} \tag{11}$$

Funkcja kryterialna $\Phi(\mathbf{w})$ jest dodatnio ważoną sumą funkcji kary $\varphi^+_{jk}(\mathbf{w})$ i $\varphi^-_{jk}(\mathbf{w})$, zdefiniowaną następująco:

$$\Phi(\mathbf{w}) = \sum_{(j,k)\in J^+} \gamma_{jk}\varphi^+_{jk}(\mathbf{w}) + \sum_{(j,k)\in J^-} \gamma_{jk}\varphi^-_{jk}(\mathbf{w}) \tag{12}$$

gdzie γ_{jk} jest dodatnim parametrem (*ceną*) związanym z *wektorem różnicowym* $\mathbf{r}_{jk} = \rho_k - \rho_j$, J^+ jest zbiorem par indeksów (j,k) wektorów ze zbioru dodatniego R^+ $((j,k) \in J^+ \Leftrightarrow \mathbf{r}_{jk} \in R^+)$, J^- jest zbiorem par indeksów (j,k) wektorów ze zbioru ujemnego R^- $((j,k) \in J^- \Leftrightarrow \mathbf{r}_{jk} \in R^-)$ (7).

Perceptronowa funkcja kryterialna używana w teorii sieci neuropodobnych i rozpoznawania obrazów ma postać podobną do $\Phi(\mathbf{w})$ (12) [2]. Funkcja kryterialna $\Phi(\mathbf{w})$ (12) jest funkcją wypukłą i odcinkowo-liniową (ang. *convex and piecewise linear - CPL*) jako suma tego typu funkcji kary $\varphi^+_{jk}(\mathbf{w})$ (10) i $\varphi^+_{jk}(\mathbf{w})$ (11). Algorytmy wymiany rozwiązań bazowych, zbliżone do algorytmów programowania liniowego, pozwalają znaleźć minimum funkcji $\Phi(\mathbf{w})$ w sposób efektywny nawet w przypadku dużych, wielowymiarowych zbiorów R^+ i R^- (7) [3]:

$$\Phi^* = \Phi(\mathbf{w}^*) = \min_{\mathbf{w}} \Phi(\mathbf{w}) \ge 0 \tag{13}$$

Optymalny wektor parametrów $\mathbf{w}^*$ oraz wartość minimalna $\Phi(\mathbf{w})^*$ funkcji kryterialnej $\Phi(\mathbf{w})$ (12) mogą być stosowane w rozwiązywaniu wielu problemów eksploracyjnej analizy danych. W szczególności wektor $\mathbf{w}^*$ pozwala wyznaczyć optymalne odwzorowanie rangowe $t = t(\mathbf{w}^*) = (\mathbf{w}^*)^T\rho$ zachowujące wszystkie implikacje rangowe (6) lub ich większość. Można wykazać słuszność poniższego Lematu [2]:

Lemat 2. *Wartość minimalna (13) funkcji kryterialnej (12) jest równa zeru,* $\Phi^* = 0$ *wtedy i tylko wtedy, gdy istnieje taki wektor parametrów* $\mathbf{w}'$*, że wszystkie implikacje rangowe (6) zachowane są przez odwzorowanie liniowe* $t = t(\mathbf{w}') = (\mathbf{w}')^T \rho$ *(5).*

Można zauważyć też, że $\Phi(\mathbf{w}^*) = 0$ wtedy i tylko wtedy, gdy istnieje taka hiperpłaszczyzna $H(\mathbf{w}')$ (9), która w pelni separuje zbiór dodatni R^+ od zbioru ujemnego R^- (7). Jeżeli hiperpłaszczyzna $H(\mathbf{w}^*)$ (9) wyznaczona przez wektor optymalny $\mathbf{w}^*$ (13) nie separuje zbiorów R^+ i R^-, to zbiory te nie są saparowalne przez żadną inną hiperpłaszczyznę $H(\mathbf{w})$ (9). Wartość minimalna $\Phi(\mathbf{w}^*)$ (13) jest wtedy większa od zera ($\Phi(\mathbf{w}^*) > 0$).

5. Wybrane problemy szeregowania zadań obliczeniowych

Wiele problemów szeregowania zadań obliczeniowych O_j może być analizowanych z wykorzystaniem zbioru wektorów zależności ρ_j (1). Załóżmy, że realizowany proces obliczeniowy podzielono (zdekomponowano) na m zadań O_j scharakteryzowanych przez wektory zależności ρ_j (1) oraz czasy obliczeń τ_j:

$$C_0 = \{O_j\}, \qquad gdzie \quad j = 1, ..., m \tag{14}$$

Jednym z podstawowych pytań jest to, czy zbiór C_0 zadań obliczeniowych reprezentowanych przez wektory zależności jest niesprzeczny i może być zrealizowany w sposób sekwencyjny? Innymi słowy, czy dany proces obliczeniowy może zostać zrealizowany jako pewna sekwencja zadań O_j (dla $j = 1, ..., m$)?

Szukając odpowiedzi na to pytanie, posłużymy się konstrukcją tzw. *warstw funkcjonalnych*. Zerowa warstwa funkcjonalna F_0 jest utworzona przez m_0 takich zadań obliczeniowych O_k ze zbioru C_0 (14), które nie zależą od żadnych innych zadań O_j, tj.:

$$F_0 = \{O_k : \ \rho_k = [0, ..., 0]^T\} \tag{15}$$

Zakładamy tu, że $m_0 > 0$.

W celu tworzenia kolejnych warstw funkcjonalnych F_n ($n > 0$) zbiór C_0 (14) zostaje zredukowany do zbioru C_1 poprzez pominięcie takich m_0 zadań, które wchodzą do zerowej warstwy funkcjonalnej F_0 (15):

$$C_1 = C_0 / F_0 \tag{16}$$

Redukcja zbioru C_0 do zbioru C_1 oznacza też redukcję tych składowych ρ_{kj} w wektorach zależności ρ_k (1), które odpowiadają pomijanym zadaniom O_j ($O_j \in F_0$). Jeżeli zbiór F_0 zawiera m_0 zadań, to zbiór C_1 zawiera $m - m_0$ zadań scharakteryzowanych wektorami zależności $\rho[m-m_0]$ o wymiarze $m-m_0$. Jeżeli $m-m_0>0$, to pierwsza warstwa funkcjonalna F_1 zostaje utworzona przez takich m_1 ($m_1>0$) zadań obliczeniowych O_k ze zbioru C_1 (16), które nie zależą od innych zadań O_j z tego zbioru, tj.:

$$F_1=\{O_k:\ O_k\in C_1,\ oraz\ \rho_k[m-m_0]=[0,...,0]^T\} \tag{17}$$

W kolejnym etapie zbiór C_1 (16) zostaje zredukowany do zbioru C_2 poprzez pominięcie takich m_1 zadań O_k, które wchodzą do pierwszej warstwy funkcjonalnej F_1 (17):

$$C_2=C_1/F_1 \tag{18}$$

Podobnie jak poprzednio, redukcji zbioru C_1 zadań O_j (18) do zbioru C_2 towarzyszy redukcja składowych ρ_{kj} odpowiadających pomijanym zadaniom oraz powstanie zredukowanych wektorów zależności $\rho_k[m-m_0-m_1]$ o wymiarze $m-m_0-m_1$. Jeżeli $m-m_0-m_1>0$, to podjęta zostaje próba utworzenia drugiej warstwy funkcjonalnej F_2 w sposób podobny do tworzenia warstwy F_1. W podobny sposób tworzone są następne warstwy funkcjonalne F_n ($n>2$) aż do osiągnięcia etapu, w którym kolejna warstwa funkcjonalna nie może być utworzona.

Dwie przyczyny mogą spowodować, że kolejna warstwa funkcjonalna F_n nie może być utworzona. Po pierwsze, warstwa funkcjonalna F_n nie może być utworzona w sytuacji, gdy zbiór C_n typu (18) jest zbiorem pustym ($C_n=\phi$), ponieważ wszystkie zadania obliczeniowe O_k zostały już zredukowane we wcześniejszych warstwach $F_{n'}$ ($n'<n$). W tym przypadku mamy równość:

$$m-m_0-...-m_{n-1}=0 \tag{19}$$

Jest to naturalne zakończenie procesu tworzenia $n-1$ warstw funkcjonalnych F_i ($i=1,...,n-1$).

Drugim powodem uniemożliwiającym utworzenie warstwy funkcjonalnej F_n może być brak takiego zadania O_k w niepustym zbiorze $C_n=C_{n-1}/F_{n-1}$ (18), dla którego zredukowany wektor zależności $\rho_k[m-m_0-...-m_{n-1}]$ jest równy wektorowi zerowemu $[0,...,0]^T$:

$$(\forall k\in I_n)\quad \rho_k[m-m_0-...-m_{n-1}]\neq[0,...,0]^T \tag{20}$$

gdzie $k\in I_n \Leftrightarrow O_k\in C_n$.

Warunek stopu (20) oznacza, że wszystkie zadania O_k należące do zredukowanego zbioru C_n są ze sobą powiązane. Powstaje pętla, która uniemożliwia wydzielenie zadania odpowiedniego do realizacji. Nie można zrealizować żadnego zadania O_k ze zbioru C_n, ponieważ realizacja każdego z tych zadań wymaga znajomości wyników pewnego innego zadania z tego zbioru. Podobny problem może się pojawić już w momencie tworzenia zerowej warstwy funkcjonalnej F_0 (15). Warunek uniemożliwiający utworzenie warstwy F_0 ma postać:

$$(\forall O_k \in C_0) \quad \rho_k[m] \neq [0, ..., 0]^T \tag{21}$$

Warunki typu (20) lub (21) oznaczają pojawienie się pętli w grafie zależności. Pojawienie się pętli w zbiorze C_n (18) oznacza, że zbiór zadań O_k z tego zbioru, reprezentowanych za pomocą zredukowanych wektorów zależności $\rho_k[m - m_0 - ... - m_{n-1}]$, jest sprzeczny i nie może być zrealizowany. Realizacja każdego zadania O_k tworzącego pętlę jest niemożliwa, ponieważ wymaga znajomości wyników pewnego innego zadania O_j ze zbioru C_n.

Algorytm 1 Algorytm pozwalający stwierdzić sprzeczność lub niesprzeczność zbioru zadań obliczenowych

1: $C_0 := \{0_j\}\ j = 1, ..., m$
2: $I_0 := \{j\}\ j = 1, ..., m$
3: $n := 0$
4: **if** $C_n = \phi$ **then**
5: KONIEC: zbiór zadań jest niesprzeczny
6: **end if**
7: **if** $(\forall k \in I_n)\ \rho_\mathbf{k} \neq [0, ..., 0]^T$ **then**
8: KONIEC: zbiór zadań jest sprzeczny
9: **end if**
10: $F_n := \{O_k : O_k \in C_n \ oraz\ \rho_\mathbf{k} = [0, ..., 0]^T$
11: $C_{n+1} := C_n / F_n$
12: $I_{n+1} := I_n / \{k : O_k \in F_n\}$
13: zredukuj składowe ρ_{kj} w wektorach zależności $\rho_\mathbf{k}$ $(O_k \in C_{n+1})$ odpowiadające pomijanym zadaniom O_j $(O_j \in F_n)$
14: $n := n + 1$
15: GOTO 4

Zauważmy, że przynależność zadań O_j i O_k do tej samej warstwy funkcjonalnej F_n wyklucza istnienie zależności $O_j \prec O_k$ (2) lub $O_k \prec O_j$ pomiędzy tymi zadaniami. Równoległa realizacja zadań obliczeniowych należących O_j do tej samej warstwy funkcjonalnej F_n daje możliwość skrócenia czasu realizacji całego procesu obliczeniowego.

Lemat 3. *Jeżeli zadania O_j i O_k należą do tej samej warstwy funkcjonalnej F_n, to nie może istnieć zależność $O_j \prec O_k$ (2) pomiędzy tymi zadaniami.*

Słuszność lematu 3 wynika bezpośrednio z opisanej powyżej konstrukcji warstw funkcjonalnych F_n, gdzie każde zadanie O_k danej warstwy jest scharakteryzowane zredukowanym wektorem zależności ρ_k o wartości zerowej ($\rho_k = [0, ..., 0]^T$. Podobnie można uzasadnić poniższą własność:

$$(\forall O_{k'} \in F_{n'})\ (\forall O_k \in F_n) \quad (O_{k'} \prec O_k \Rightarrow n' < n) \tag{22}$$

Zgodnie z powyższą własnością, istnienie zależności $O_{k'} \prec O_k$ w strukturze warstw funkcjonalnych $F_{n'}$ i F_n wyznacza taki porządek, że warstwa F_n znajduje się wyżej niż warstwa $F_{n'}$. Zadanie O_k z warstwy wyższej nie może poprzedzać zadania $O_{k'}$ z warstwy niższej.

Lemat 4. *Zadania obliczeniowe ze zbioru C_0 (14) mogą być zrealizowane wtedy i tylko wtedy, gdy podczas wydzielania warstw funkcjonalnych F_n nie zachodzą warunki typu (20) lub (21) wskazujące na „zapętlenie się” procesu obliczeniowego.*

Istnienie pętli w zbiorze C_0 (14) m zadań obliczeniowych ma związek z minimalną wartością $\Phi(\mathbf{w}^*[m])$ (13) funkcji kryterialnej $\Phi(\mathbf{w}[m])$ (12):

Twierdzenie 1. *Zbiór C_0 opisuje niesprzeczny zestaw m zadań obliczeniowych O_k wtedy i tylko wtedy, gdy wartość minimalna $\Phi(\mathbf{w}^*[m])$ funkcji kryterialnej $\Phi(\mathbf{w}[m])$ określonej na bazie wektorów zależności $\rho_k[m]$ (1) reprezentujących te zadania jest równa zeru ($\Phi(\mathbf{w}^*[m]) = 0$).*

Dowód: W dowodzie posłużymy się opisaną powyżej konstrukcją warstw funkcjonalnych F_n. Załóżmy, że w trakcie konstrukcji nie są spełnone warunki typu (20) lub (21). W tym przypadku wszystkie zadania obliczeniowe ze zbioru C_0 (14) mogą być rozłożone w warstwy funkcjonalne F_n. W rezultacie wszystkie zadania mogą być zrealizowane w sposób sekwencyjny, zgodny z podziałem na warstwy F_n.

Dla przeprowadzenia dowodu dokonamy indeksowania zadań obliczeniowych O_j ze zbioru C_0 zgodnie z podziałem na warstwy funkcjonalne F_n (*indeksowanie kanoniczne*). Przyjmiemy, że m_0 zadań O_j z warstwy F_0 ma najniższe indeksy j ($1 \leq j \leq m_0$), m_1 zadań O_j z warstwy F_1 ma indeksy j z przedziału ($m_0 + 1 \leq j \leq m_0 + m_1$), itd. Przy takim sposobie indeksowania zadań obliczeniowych O_j wektory zależności $\rho_j = [\rho_{j1}, ..., \rho_{jm}]^T$ o m składowych binarnych ρ_{ji} mają następującą strukturę: m_0 pierwszych wektorów ρ_j ($1 \leq j \leq m_0$) ma wszystkie m składowych ρ_{ji} równe zeru ($(\forall j \in \{1, ..., m_0\})\ (\forall i \in \{1, ..., m\})\ \rho_{ji} = 0$). Kolejne m_1 wektorów ρ_j, odpowiadających zadaniom O_j z warstwy F_1, ma składowe ρ_{ji} o indeksach i większych niż m_0 równe zeru ($(\forall j \in \{m_0 + 1, ..., m_0 + m_1\})\ (\forall i \in \{m_0 + 1, ..., m\})\ \rho_{ji} = 0$). Podobnie m_2 wektorów ρ_j odpowiadających zadaniom O_j

z warstwy F_2 ma składowe ρ_{ji} o indeksach i większych niż $m_0 + m_1$ równe zeru $((\forall j \in \{m_0 + m_1 + 1, ..., m_0 + m_1 + m_2\})\ (\forall i \in \{m_0 + m_1 + 1, ..., m\})\ \rho_{ji} = 0)$. Podobną strukturę mają wektory zależności ρ_j z kolejnych wyższych warstw. Rozumowanie to jest oparte na właściowści określonej w relacji (22).

Weźmy wektor parametrów $\mathbf{w}'[m]$ o poniższej strukturze:

$$\mathbf{w}'[m] = [c_1, ..., c_1, c_2, ..., c_2, ..., c_n, ..., c_n]^T \tag{23}$$

Pierwsze m_0 składowych wektora $\mathbf{w}'[m]$ ma wartość c_1 ($c_1 > 0$, np. $c_1 = 1$) i odpowiada zadaniom z warstwy F_0 (15). Kolejne m_1 składowych wektora $\mathbf{w}'[m]$ ma wartość c_2 ($c_2 > c_1$) itd. Ostatnie m_n składowych wektora $\mathbf{w}'[m]$ ma wartość c_n ($c_n > c_{n-1}$).

Parametry c_i (23) można wybrać tak, by wszystkie implikacje rangowe (6) były zachowane przez odwzorowanie liniowe $t_j = (\mathbf{w}'[m])^T \rho_j[m]$. Np. odpowiednio duża wartość parametru c_k odpowiadającego warstwie F_k zapewnia, że wszystkie zależności $O_{k'} \prec O_k$, gdzie $k' < k$, są reprezentowane przez nierówności (6): $(\mathbf{w}'[m])^T \rho_{k'}[m] < (\mathbf{w}'[m])^T \rho_k[m]$.

Jak wynika z lematu 2, wartość $\Phi(\mathbf{w}^*[m])$ (13) jest równa zeru ($\Phi(\mathbf{w}^*[m] = 0$) wtedy i tylko wtedy, gdy wszystkie implikacje rangowe (6) są zachowane przez odwzorowanie liniowe $t_j = (\mathbf{w}'[m])^T \rho_j[m]$.

Twierdzenie 1 wskazuje możliwość badania niesprzeczności zestawu zadań obliczeniowych O_j (15) schrakteryzowanych przez wektory zależności $\rho_j[m]$ (1) na podstawie sprawdzenia warunku, czy wartość minimalna $\Phi(\mathbf{w}^*[m])$ (13) funkcji kryterialnej $\Phi(\mathbf{w}[m])$ (12) jest równa zeru.

6. Możliwości wykorzystania modeli rangowych w modyfikacji zadań obliczeniowych

Znajomość zestawu m zadań obliczeniowych (15) scharakteryzowanych przez wektory zależności $\rho_j[m]$ (1) pozwala zdefiniować zbiory $R^+[m]$ i $R^-[m]$ (7) złożone z wektorów różnicowych $\mathbf{r}_{jk}[m] = \rho_k[m] - \rho_j[m]$, a następnie funkcję kryterialną $\Phi(\mathbf{w}[m])$ (16). Minimalizacja funkcji kryterialnej $\Phi(\mathbf{w}[m])$ daje odpowiedź na pytanie czy zestaw zadań jest niesprzeczny (Tw. 1). Zestaw m zadań obliczeniowych scharakteryzowanych przez wektory zależności $\rho_j[m]$ (1) jest sprzeczny wtedy i tylko wtedy, gdy wartość minimalna $\Phi(\mathbf{w}^*[m])$ (13) jest większa od zera ($\Phi(\mathbf{w}^*[m]) > 0$). Ma to miejsce w sytuacji, gdy zbiory $R^+[m]$ i $R^-[m]$ (7) nie są liniowo separowalne (8). W takim przypadku część wektorów różnicowych $\mathbf{r}_{jk}[m] = \rho_k[m] - \rho_j[m]$ usytuowana jest po niewłaściwej stronie hiperpłaszczyny $H(\mathbf{w}^*[m])$ (9) wyznaczonej w m-wymiarowej przestrzeni zależności przez wektor optymalny $\mathbf{w}^*[m]$ (13).

Część wektorów $\mathbf{r}_{jk}[m]$ należących do zbioru $R^+[m]$ usytuowana jest niewłaściwie po ujemnej stronie hiperpłaszczyzny $H(\mathbf{w}^*[m])$ $(\mathbf{w}^*[m]\mathbf{r}_{jk} < 0)$. Istnieją także takie wektory $\mathbf{r}_{jk}[m]$ ze zbioru $R^-[m]$, które są niewłaściwie usytuowane po dodatniej stronie $H(\mathbf{w}^*[m])$ $(\mathbf{w}^*[m]\mathbf{r}_{jk} > 0)$. W problemach rozpoznawania obrazów znany jest fakt, że dwa zbiory liniowo nieseparowalne można doprowadzić do liniowej separowalności poprzez redukcję przynajmniej części elemetów niewłaściwie usytuowanych względem hiperpłaszczyzny optymalnej $H(\mathbf{w}^*[m])$ [5]. Podobne techniki redukcji można stosować w szeregowaniu zadań obliczeniowych.

Zgodnie z opisaną wcześniej wieloetapową procedurą wydzielana jest warstwa funkcjonalna F_0 (15) a następnie kolejne warstwy F_n. Wydzielenie kolejnej warstwy F_n wiąże się z redukcją $C_n = C_{n-1}/F_{n-1}$ (18) zbioru zadań i sprawdzaniu czy zachodzi warunek stopu (20). Warunek stopu w warstwie zerowej ma podobną postać (21). Warunek stopu wskazuje, że żadne z m zadań O_k ze zbioru C_n nie może być zrealizowane, ponieważ wymaga to znajomości wyników pewnego innego zadania ze zbioru C_n. Sposobem na przerwanie tego typu wzjemnych zależności może być taka modyfikacja jednego z zadań O_k ze zbioru C_n, by jego wykonanie nie zależało od żadnego innego zadania O_j ze zbioru C_n. Zredukowany wektor zależności $\rho_k[m]$ zmodyfikowanego zadania O'_k powinien być równy wektorowi zerowemu $[0,...,0]^T$. Modyfikacja zadania O_k prowadząca do zerowania się wektora zależności $\rho_k[m']$ jest na ogół możliwa, ale może wymagać przedefiniowania nie tylko jednego, lecz większej liczby zadań obliczeniowych, co może być obarczone dużymi kosztami.

W poszukiwaniu najbardziej efektywnego sposobu modyfikacji zadań O_k ze sprzecznego zbioru C_n można posłużyć się minimalną wartością $\Phi(\mathbf{w}^*[m])$ (13) funkcji kryterialnej $\Phi(\mathbf{w}[m])$ (12), określonej na bazie wektorów zależności $\rho_k[m]$ (1) reprezentujących zadania O_k ze zbioru C_n. Bierzemy tu pod uwagę taką modyfikację zadań $O_k \to O'_k$, którym towarzyszy zerowanie się zredukowanych wektorów zależności $\rho_k[m]$:

$$\rho_k[m] \to [0,...,0]^T \tag{24}$$

Zauważmy, że jeżeli modyfikowanie zadania O_k ze sprzecznego zbioru C_n następuje po wcześniejszym wydzieleniu pewnych warstw funkcjonalnych F_n, to warunek (24) nie ogranicza zależności zadania O_k od innych zadań z tych warswtw.

Wybór zadania O_k do modyfikacji można oprzeć na kryterium maksymalnego spadku $\triangle_k\Phi(\mathbf{w}^*[m])$ minimalnej wartości $\Phi(\mathbf{w}^*[m])$ (13) funkcji kryterialnej $\Phi(\mathbf{w}[m])$ (12) w wyniku modyfikacji (24) wektora zależności $\rho_k[m]$ (1) zadania O_k:

$$\triangle_k\Phi(\mathbf{w}^*[m]) = \max_{j \in I_n} \triangle_j\Phi(\mathbf{w}^*[m]) \tag{25}$$

gdzie $j \in I_n \Leftrightarrow O_j \in C_n$

Zgodnie z powyższym kryterium modyfikowania, do modyfikacji wybieramy takie zadanie, które daje największy spadek minimalnej wartości funkcji kryterialnej określonej na zredukowanym zbiorze C_n.

Można założyć, że obliczanie potencjalnych spadków $\triangle_j\Phi(\mathbf{w}^*[m])$ minimalnej wartości $\Phi(\mathbf{w}^*[m])$ (13) funkcji kryterialnej $\Phi(\mathbf{w}[m])$ (12) jest stosunkowo tanie w stosunku do kosztów δ_j ($\delta_j > 0$) rzeczywistej modyfikacji poszczególnych zadań obliczeniowych O_j.

Jeżeli znane są koszty δ_j modyfikacji poszczególnych zadań obliczeniowych O_j, to kryterium wyboru zadania O_k do modyfikacji może uwzględniać także wielkości δ_j:

$$\triangle_k\Phi(\mathbf{w}^*[m])/\delta_k = \max_{O_j \in C_n} \triangle_j\Phi(\mathbf{w}^*[m])/\delta_j \tag{26}$$

Opisana powyżej procedura pozwala usuwać sprzeczność́i z zestawów zadań obliczeniowych O_j poprzez modyfikację wybranych zadań O_k.

Innym ważnym w praktyce problemem jest zagadnienie redukcji niepotrzebych zadań obliczeniowych O_j tworzących zbiór C_0 (14). W pewnych przypadkach możemy być zainteresowani w jak najszybszym uzyskaniu wyniku określonego zadania O_k. Uzyskanie wyniku danego zadania O_k wymaga wykonania pewnych zadań O_j, lecz wykonywanie innych zadań ze zbioru C_0 może nie być konieczne. Zidentyfikowanie takich niekoniecznych zadań O_j pozwala je pominąć i w rezultacie skrócić czas obliczeń.

Załóżmy, że zbiór C_0 zadań O_j scharakteryzowych przez wektory zależności ρ_j (1) i czasy realizacji τ_j jest niesprzeczny. W tym przypadku wszystkie zadania O_j mogą być przyporządkowane poszczególnym warstwom funkcjonalnym F_n (19). Dogodnie jest posługiwać się wtedy indeksowaniem kanonicznym opisanym w dowodzie Twierdzenia 1. Redukcja zadań O_j oznacza też zmniejszanie wymiaru wektorów zależności $\rho_j[m]$ (1).

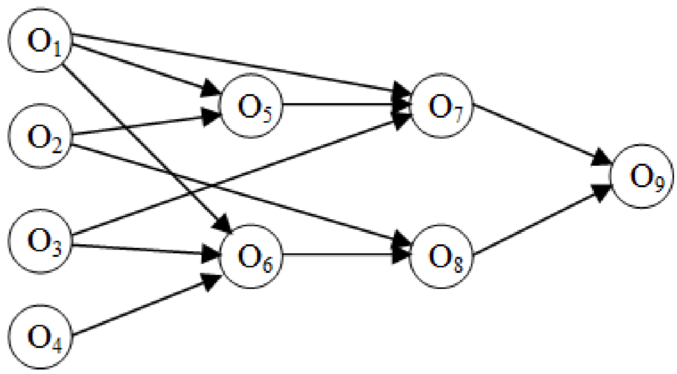
Jeżeli zadania O_j zostały przyporządkowane poszczególnym warstwom funkcjonalnym F_n i wybrane zadanie O_k znalazło się w warstwie $F_{n'}$, to wszystkie zadania z warstw wyższych F_n ($n > n'$), jak również różne od O_k zadania z warstwy $F_{n'}$ mogą zostać zredukowane (pominięte). W rezultacie takiej redukcji powstaje zbiór zredukowany C'_0. Zbiór C'_0 może być w pewnych okolicznościach dodatkowo zredukowany. W tym celu sprawdzamy czy w zbiorze C'_0 znajdzie się takie zadanie $O_{j'}$, które nie poprzedza żadnego innego zadania z tego zbioru. Jeżeli takie zadanie zostanie znalezione, to zadanie to usuwamy ze zbioru C'_0 i powtórnie poszukujemy w ostatnio zredukowanym zbiorze C'_0 zadania $O_{j''}$, które nie poprzedza żadnego innego zadania z tego zbioru. Jeżeli zadanie $O_{j''}$ o takiej właściowości bdzie znalezione, to usuwamy je ze zbioru C_0'. Poszukiwania niekoniecznych zadań O_j i ich redukcje powtarzamy do momentu, gdy w zbiorze C_0' nie będzie elementów nadających się do redukcji.

7. Wyniki eksperymentów

Przy zastosowaniu opisanych wczesniej metod wykonane zostały dwa eksperymenty, sprawdzające w praktyce słuszność zaproponowanych rozwiązań teoretycznych. W obu eksperymentach użyto sztucznie wygenerowanych sekwencji zadań obliczeniowych.

7.1 Eksperyment 1 - Sekwencja niesprzeczna

W pierwszym eksperymencie sekwencja zadań obliczeniowych bya dobrana tak, aby zdekomponowany proces obliczeniowy był niesprzeczny, czyli aby nie występowały w nim zapętlenia. Rysunek 1 pokazuje kolejność wykonywania zadań. Strzałka od zadania O_j do zadania O_k wskazuje na zależność zadania O_k od zadania O_j. Pomiędzy zadaniami O_j i O_k zachodzi relacja rangowa $O_j \prec O_k$.



Rys. 1. Sekwencja zadań użytych w eksperymencie 1

Otrzymano następujące wektory zależności ρ_j ($j = \{1, ..., 9\}$):

$$\begin{aligned}
\rho_1 &= [0,0,0,0,0,0,0,0,0]^T \\
\rho_2 &= [0,0,0,0,0,0,0,0,0]^T \\
\rho_3 &= [0,0,0,0,0,0,0,0,0]^T \\
\rho_4 &= [0,0,0,0,0,0,0,0,0]^T \\
\rho_5 &= [1,1,0,0,0,0,0,0,0]^T \\
\rho_6 &= [1,0,1,1,0,0,0,0,0]^T \\
\rho_7 &= [1,0,1,0,1,0,0,0,0]^T \\
\rho_8 &= [0,1,0,0,0,1,0,0,0]^T \\
\rho_9 &= [0,0,0,0,0,0,1,1,0]^T
\end{aligned} \tag{27}$$

Na podstawie relacji rangowych pomiędzy zadaniami i ich wektorów zależności wyznaczono zbiory R^+ i R^- (7) wektorów różnicowych $\mathbf{r}_{jk}$:

$$\begin{aligned}
\mathbf{r}_{1,5} &= [1,1,0,0,0,0,0,0,0]^T \\
\mathbf{r}_{1,6} &= [1,0,1,1,0,0,0,0,0]^T \\
\mathbf{r}_{1,7} &= [1,0,1,0,1,0,0,0,0]^T \\
\mathbf{r}_{2,5} &= [1,1,0,0,0,0,0,0,0]^T \\
\mathbf{r}_{2,8} &= [0,1,0,0,0,1,0,0,0]^T \\
\mathbf{r}_{3,6} &= [1,0,1,1,0,0,0,0,0]^T \\
\mathbf{r}_{3,7} &= [1,0,1,0,1,0,0,0,0]^T \\
\mathbf{r}_{4,6} &= [1,0,1,1,0,0,0,0,0]^T \\
\mathbf{r}_{5,7} &= [0,-1,1,0,1,0,0,0,0]^T \\
\mathbf{r}_{6,8} &= [-1,1,-1,-1,0,1,0,0,0]^T \\
\mathbf{r}_{7,9} &= [-1,0,-1,0,-1,0,1,1,0]^T \\
\mathbf{r}_{8,9} &= [0,-1,0,0,0,-1,1,1,0]^T \\
\\
R^+ &= \{\mathbf{r}_{1,5}, \mathbf{r}_{1,6}, \mathbf{r}_{1,7}, \mathbf{r}_{2,5}, \mathbf{r}_{2,8}, \mathbf{r}_{3,6}, \mathbf{r}_{3,7}, \mathbf{r}_{4,6}, \mathbf{r}_{5,7}, \mathbf{r}_{6,8}, \mathbf{r}_{7,9}, \mathbf{r}_{8,9}\} \\
R^- &= \phi
\end{aligned} \tag{28}$$

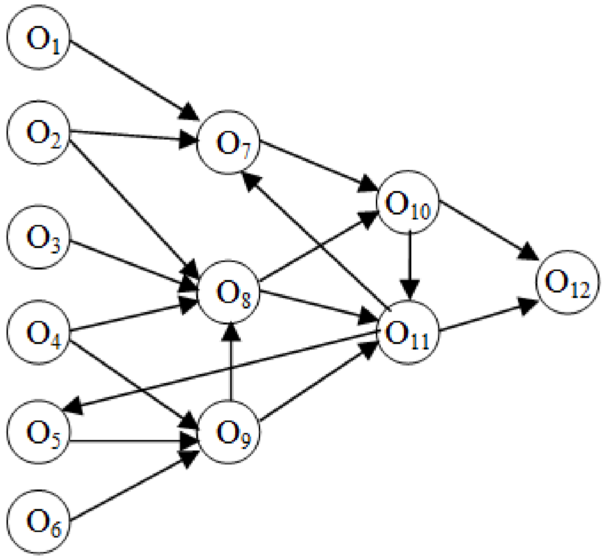
Poprzez minimalizację wartości funkcji kryterialnej $\Phi(\mathbf{w})$ (12) wyznaczono optymalny wektor parametrów $\mathbf{w}^*$:

$$\mathbf{w}^* = [-1,0,-1,0,0,0,-3,-4,0,0]^T \tag{29}$$

Wartość minimalna funkcji kryterialnej $\Phi(\mathbf{w}^*)$ (13) przyjmuje wartość zero, więc zgodnie z twierdzeniem 1 zestaw zadań jest niesprzeczny i może być realizowany w założonej sekwencji.

7.2 Eksperyment 2 - Sekwencja sprzeczna

Drugi eksperyment był identyczny z pierwszym. Została w nim jedynie użyta sekwencja zadań, w której występują pętle.



Rys. 2. Sekwencja zadań użytych w eksperymencie 2

Wyznaczone zostały wektory zależności ρ_j $(j = \{1,...,12\})$ i zbiory R^+ i R^- (7) wektorów różnicowych $\mathbf{r}_{jk}$.

$$R^+ = \{\mathbf{r}_{1,7}, \mathbf{r}_{2,7}, \mathbf{r}_{2,8}, \mathbf{r}_{3,8}, \mathbf{r}_{4,8}, \mathbf{r}_{4,9}, \mathbf{r}_{5,9}, \mathbf{r}_{6,9}, \mathbf{r}_{7,10}, \mathbf{r}_{8,10}, \mathbf{r}_{8,11}, \mathbf{r}_{9,11}, \mathbf{r}_{10,11}, \mathbf{r}_{10,12}, \mathbf{r}_{11,12}\}$$
$$R^- = \{\mathbf{r}_{5,11}, \mathbf{r}_{7,11}, \mathbf{r}_{8,9}\} \tag{30}$$

Poprzez minimalizację wartości funkcji kryterialnej $\Phi(\mathbf{w})$ (12) wyznaczono optymalny wektor parametrów $\mathbf{w}^*$:

$$\mathbf{w}^* = [0,0,0,0,-1,0,-3,0,-2,0,-3.5,0]^T \tag{31}$$

Wartość minimalna funkcji kryterialnej $\Phi(\mathbf{w}^*)$ (13) przyjmuje wartość 0.250367, więc zgodnie z przewidywaniami otrzymano odpowiedź, że sekwencja zadań nie może być zrealizowana w takiej kolejności. Stosując mechanizm opisany w rozdziale 6 można wyznaczyć zadanie do modyfikacji i doprowadzić do niesprzeczności sekwencji.

8. Uwagi końcowe

Regresja rangowa może być użytecznym narzędziem analizy dużych systemów obliczeniowych. Budowa modeli rangowych do celów analizy systemów obliczeniowych opiera się na reprezentacji zadań obliczeniowych O_j za pomocą wektorów zależności ρ_j (1), które opisują wzajemne zależności $O_j \prec O_k$ (2) pomiędzy tymi zadaniami. Modele rangowe buduje się w formie takich odwzorowań liniowych z wielowymiarowej przestrzeni zależności na przestrzeń jednowymairową, które w maksymalnym stopniu zachowują (6) te zależności.

Konstrukcja odwzorowań rangowych opiera się na wyznaczaniu hiperpłaszczyzny $H(\mathbf{w}')$ (9) oddzielającej w możliwie najlepszy sposób dwa zbiory R^+ i R^- (7). Metoda wyznaczania hiperpłaszczyzny $H(\mathbf{w}')$ (9) bazuje na minimalizacji wypukłej i odcinkowo-liniowej (ang. *convex and piecewise linear - CPL*) funkcji kryterialnej $\Phi(\mathbf{w})$ (12). Zostało wykazane (Tw. 1), że wartość minimalna $\Phi(\mathbf{w}^*[m])$ (13) funkcji kryterialnej $\Phi(\mathbf{w}[m])$ (12) określonej na bazie m wektorów zależności ρ_j (1) reprezentujących zadania obliczeniowe O_j jest równa zeru ($\Phi(\mathbf{w}^*[m]) = 0$) wtedy i tylko wtedy, gdy zbiór C_0 (14) zawiera niesprzeczny zestaw zadań.

Minimalizacja funkcji kryterialnej $\Phi(\mathbf{w}[m])$ (12) pozwala nie tylko wykrywać sprzeczności w zestawach zadań obliczeniowych O_j, ale może być również użyteczna w wybieraniu zadań do modyfikacji mających na celu efekywne usunięcie tych sprzeczności.

Literatura

[1] Bobrowski, L.: Modele szeregowania zadań obliczeniowych wykorzystujące regresję rangową, „Symulacja w Badaniach i Rozwoju”, PTSK, Warszawa, 2007.

[2] Bobrowski, L.: Linear ranked regression - designing principles, CORES05, IV International Conference on Computer Recognition Systems, Advances in Soft Computing, Springer, Berlin, 2005.

[3] Bobrowski, L., Niemiro, W.: A method of synthesis of linear discriminant function in the case of nonseparabilty, Pattern Recognition 17, pp.205-210, 1984.

[4] Błażewicz, J., Ecker, K., Pesh, E., Schmidt, G., Węglarz, J.: Scheduling Computer and Manufacturing Process, Springer, Berlin, 1966.

[5] Duda, O.R., Heart, P.E., Stork, D.G.: Pattern Classification, Wydanie drugie, zmienione, John Wiley & Sons, 2001.

[6] Janiak, A. (Ed.): Scheduling in computer and manufacturing systems, WKiŁ, 2006.

[7] Smutnicki, C.: Algorytmy szeregowania Exit 2002.

SCHEDULING BASED ON RANKED REGRESSION MODELS

Abstract: The issues of scheduling of tasks are found, among other things, in connection with the problems of realizeable of big computing processes and optimisation of them. The ranked regresion methods can be used to determine of this kind of problems. Separate computing tasks are characterized by multidimensional vectors of dependences in order to form the ranked regresion models . The vectors of dependences allow to state whether particular task can be realised only when certain other tasks have realised before. The ranked regresion includes the designing of such linear transformations from the multidimensional space of dependences to unidimensional space (time line), which reflect the dependences beetwen task as well as possible.

Keywords: scheduling of the computing tasks, ranked model, convex and piecewise linear (CPL) criterion functions

Praca wspierana częściowo przez projekt KBN 3T11F011 30, projekty W/WI/1/2008 i S/WI/2/2008 Politechniki Białostockiej.