Академия информатизации образования

Академия компьютерных наук

Институт управления образованием РАО

Ассоциация электронного обучения

ТРУДЫ МЕЖДУНАРОДНОЙ НАУЧНО-ПРАКТИЧЕСКОЙ КОНФЕРЕНЦИИ

«ИНФОРМАТИЗАЦИЯ ОБРАЗОВАНИЯ–2016»

14 – 17 июня 2016 г.
г. Сочи

Москва
2016

Вычислительные алгоритмы декомпозиции в задачах оптимизации на графах и сетях с дополнительными ограничениями

Пилипчук Л.А., Пилипчук А.С.
Белорусский государственный университет

Рассматриваются вычислительные алгоритмы и технологии декомпозиции в задачах оптимизации на графах и сетях с дополнительными ограничениями.

Ключевые слова: граф, декомпозиция, поток, дополнительные ограничения, целевая функция, дерево, корневое дерево, оптимальное решение.

COMPUTATIONAL DECOMPOSITION ALGORITHMS IN OPTIMIZATION PROBLEMS ON GRAPHS AND NETWORKS

WITH ADDITIONAL RESTRICTIONS

Lyudmila A.Pilipchuk, Andrei S. Pilipchuk
Belarusian State University

Computational decomposition algorithms and technologies in optimization problems on graphs and networks with additional restrictions are considered.

Keywords: graph, decomposition, flow, additional constraints, objective function, tree, rooted tree, optimal solution

Рассмотрим конечную ориентированную сеть $G=(I,U)$, $|U|>>|I|$ с множеством узлов I и множеством дуг U, без кратных дуг и петель. Пусть $I^* \subseteq I$ - множество узлов с переменными интенсивностями $\pm x_i$, $I^* \neq \varnothing$, $sign(i)=1$, если $i \in I^*_+$, $sign(i)=-1$, если $i \in I^*_-$, $I^*_+, I^*_- \subseteq I^*, I^*_+ \bigcap I^*_- = \varnothing$. $I \setminus I^*$ - множество узлов с постоянными интенсивностями a_i, $i \in I \setminus I^*$. Введем для узлов $i \in I^*$ характеристику $\tilde{n}_i$, которая для узлов из множества I^*_+ означает затраты, связанные с увеличением производства на единицу продукта, для узлов из множества I^*_- означает затраты на хранение единицы продукта. Остальные характеристики оставим традиционными: d_{ij} - пропускная способность дуги (i,j); x_{ij} - дуговой поток; c_{ij} - стоимость перевозки единицы потока по дуге (i,j), $(i,j) \in U$, $I^+_i(U)=\{j:(i,j)\in U\}$, $I^-_i(U)=\{j:(j,i)\in U\}$. На сети G рассмотрим математическую модель экстремальной задачи вида

$$\phi(x)=\sum_{(i,j)\in U} c_{ij}x_{ij}+\sum_{i\in I^*} c_i x_i \to \min, \tag{1}$$

$$\sum_{j\in I^+_i(U)} x_{ij}-\sum_{j\in I^-_i(U)} x_{ji}=\begin{cases} x_i sign(i), i\in I^*, \\ a_i, i\in I\setminus I^* \end{cases} \tag{2}$$

$$\sum_{(i,j)\in U} \lambda^p_{ij}x_{ij}+\sum_{i\in I^*}\lambda^p_i x_i=\beta^p, p=\overline{1,q}, \tag{3}$$

$$0\le x_{ij}\le d_{ij}, (i,j)\in U,\ b_{*i}\le x_i\le b^*_i, i\in I^*. \tag{4}$$

Вектор $x=(x_{ij},(i,j)\in U, x_i, i\in I^*)$ - план, $x\in X$, (X - множество планов), если на нем выполняются ограничения (2) - (4) задачи. План $x^0\in X$ оптимален, если $c'x^0=\min c'x, x\in X$, $c=(c_{ij},(i,j)\in U, c_i, i\in I^*)$. При заданном $\varepsilon\geq 0$, ε - оптимальный план $x^\varepsilon=(x^\varepsilon{}_{ij},(i,j)\in U, x_i^\varepsilon, i\in I^*)$ определяется неравенством $c'x^\varepsilon-c'x^0\leq\varepsilon, x^\varepsilon\in X$. Опорный план - пара $\{x,K\}$ из плана и опоры $K=\left\{U_K, I_K^*\right\}$ [1]. Совокупность множеств $K=\left\{U_K, I_K^*\right\}$ может быть представлена в виде объединения непересекающихся совокупностей множеств $R=\left\{U_R, I_R^*\right\}$, $W=\left\{U_W, I_W^*\right\}$, $U_W\subseteq U\setminus U_R, I_W^*\subseteq I^*\setminus I_R$ и $U_K=U_R\bigcup U_W$, $U_R\bigcap U_W=\varnothing$, $I_K^*=I_R^*\bigcup I_W^*$, $I_R^*\bigcap I_W^*=\varnothing$ со свойствами, приведенными в [1]. Опорный план $\{x,K\}$- невырожденный, если он удовлетворяет следующим условиям:

$$0<x_{ij}<d_{ij},(i,j)\in U_K,\ b_{*i}\leq x_i\leq b_i^*, i\in I_K^*. \qquad (5)$$

Наряду с планом x рассмотрим план $\overline{x}$, $\Delta x=\overline{x}-x$. Из [1] следует, что для приращения $\Delta x=(\Delta x_{ij},(i,j)\in U;\Delta x_i, i\in I^*)$ выполняются соотношения:

$$\Delta x_{ij}=\sum_{(\tau,\rho)\in U\setminus U_R}\delta_{ij}^{\tau\rho}\Delta x_{\tau\rho}+\sum_{\gamma\in I^*\setminus I_R^*}\delta_{ij}^{\gamma}\Delta x_\gamma,(i,j)\in U_R$$

$$\Delta x_i=\sum_{(\tau,\rho)\in U\setminus U_R}\delta_i^{\tau\rho}\Delta x_{\tau\rho}+\sum_{\gamma\in I^*\setminus I_R^*}\delta_i^{\gamma}\Delta x_\gamma, i\in I_R^*, \qquad (6)$$

$$\sum_{(\tau,\rho)\in U\setminus U_R}\Lambda_{\tau\rho}^p\Delta x_{\tau\rho}+\sum_{\gamma\in I^*\setminus I_R^*}\Lambda_\gamma^p\Delta x_\gamma=0,\ p=\overline{1,q}, \qquad (7)$$

где детерминанты [1] структур, порожденных элементами множеств W, $N=\{U_N, I_N^*\}$, $U_N=U\setminus U_K, I_N^*=I^*\setminus I_K$ вычисляются следующим образом:

$$\begin{aligned}\Lambda_{\tau\rho}^p&=\sum_{(i,j)\in U_R}\lambda_{ij}^p\delta_{ij}^{\tau\rho}+\sum_{i\in I_R^*}\lambda_i^p\delta_i^{\tau\rho}+\lambda_{\tau\rho}^p,\\ \Lambda_\gamma^p&=\sum_{(i,j)\in U_R}\lambda_{ij}^p\delta_{ij}^{\gamma}+\sum_{i\in I_R^*}\lambda_i^p\delta_i^{\gamma}+\lambda_\gamma^p\end{aligned} \qquad (8)$$

Определим векторы-столбцы:

$\Delta x_W = (\Delta x_{ij}, (i,j) \in U_W; \Delta x_i, i \in I_W^*)$,

$\Delta x_R = (\Delta x_{ij}, (i,j) \in U_R; \Delta x_i, i \in I_R^*)$,

$\Delta x_N = (\Delta x_{ij}, (i,j) \in U_N; \Delta x_i, i \in I_N^*)$,

Определим матрицы:

$\Lambda_W = (\Lambda_{W_1}, \Lambda_{W_2})$,

$\Lambda_{W_1} = (\Lambda_{\tau\rho}^p, p = \overline{1,q}; (\tau,\rho) \in U_W)$'

$\Lambda_{W_2} = (\Lambda_\gamma^p, p = \overline{1,q}, \gamma \in I_W^*)$.

$S_W = (\delta_{\tau\rho}, (\tau,\rho) \in U_W; \delta_\gamma, \gamma \in I_W^*)$,

$S_N = (\delta_{\tau\rho}, (\tau,\rho) \in U_N; \delta_\gamma, \gamma \in I_N^*)$.

Матрицы Λ_W и Λ_N состоят из детерминантов структур [1], порожденных элементами множеств W, $N = \{U_N, I_N^*\}$, где $U_N = U \setminus U_K, I_N^* = I^* \setminus I_K$.

Столбцы матриц $S_W = (\delta_{\tau\rho}, (\tau,\rho) \in U_W; \delta_\gamma, \gamma \in I_W^*)$ и $S_N = (\delta_{\tau\rho}, (\tau,\rho) \in U_N; \delta_\gamma, \gamma \in I_N^*)$ в совокупности, составляют базис пространства решений системы (2) [1].

Совокупность множеств $W = \left\{U_W, I_W^*\right\}$, на основе которой построены компоненты вектора Δx_W, выбрана таким образом, чтобы $\left|\Lambda_W\right| \neq 0$. Поскольку, K – опора, то, $\left|\Lambda_W\right| \neq 0$ [1] и из (8) однозначно вычисляется вектор Δx_W:

$$\Delta x_W = -\Lambda_W^{-1} \Lambda_N \Delta x_N . \qquad (9)$$

Используя соотношения (6), вычислим приращение целевой функции:

$$\Delta\phi(x) = \sum_{(i,j)\in U} c_{ij}\Delta x_{ij} + \sum_{i\in I^*} c_i \Delta x_i = \sum_{(\tau,\rho)\in U \setminus U_R} \Delta_{\tau\rho}\Delta x_{\tau\rho} + \sum_{\gamma \in I^* \setminus I_R^*} \Delta_\gamma \Delta x_\gamma,$$

$$\Delta_{\tau\rho} = \sum_{(i,j)\in U_R} c_{ij}\delta_{ij}^{\tau\rho} + \sum_{i\in I_R^*} c_i \delta_i^{\tau\rho} + c_{\tau\rho},$$

$$\Delta_\gamma = \sum_{(i,j)\in U_R} c_{ij}\delta_{ij}^{\gamma} + \sum_{i\in I_R^*} c_i \delta_i^{\gamma} + c_\gamma .$$

Обозначим $r^l = \Delta_W^l \Lambda_W^{-1}, r = (r_1, r_2 ... r_q)$,

$\tilde{\Delta}_N = (\tilde{\Delta}_{\tau\rho}, (\tau,\rho) \in U_N; \tilde{\Delta}_\gamma, \gamma \in I_N^*)$,

$$\tilde{\Delta}_{\tau\rho} = \Delta_{\tau\rho} - \sum_{p=1}^{q} r_p \Lambda^p_{\tau\rho}, \quad (\tau,\rho) \in U_N, \quad \tilde{\Delta}_\gamma = \Delta_\gamma - \sum_{p=1}^{q} r_p \Lambda^p_\gamma,$$
$\gamma \in I^*_N$.

С учетом (9), приращение целевой функции имеет следующий вид

$$\Delta\phi(x) = \sum_{(\tau,\rho)\in U_N} \tilde{\Delta}_{\tau\rho} \Delta x_{\tau\rho} + \sum_{\gamma \in I^*_N} \tilde{\Delta}_\gamma \Delta x_\gamma . \qquad (10)$$

С использованием результатов работы [2] доказана теорема

Теорема (критерий оптимальности). Соотношения

$$\tilde{\Delta}_{ij} x_{ij} = \min_{0\le\omega\le d_{ij}} \tilde{\Delta}_{ij}\omega, (i,j) \in U_N, \ \tilde{\Delta}_i x_i = \min_{b_{*i}\le\upsilon\le b^*_i} \tilde{\Delta}_i \upsilon, i \in I^*_N, \ (11)$$

достаточны, а в случае выполнения условий (5) и необходимы для оптимальности опорного плана $\{x, K\}$.

$$u_i - u_j + \sum_{p=1}^{q} \lambda^p_{ij} r_p = c_{ij}, \ (i,j) \in U_K,$$
$$-u_i sign[i] + \sum_{p=1}^{q} \lambda^p_i r_p = c_i, \ \ i \in I^*_K, \qquad (12)$$

По известным потенциалам вычисляются значения величин оценок:

$$\tilde{\Delta}_{ij} = c_{ij} - u_i + u_j - \sum_{p=1}^{q} \lambda^p_{ij} r_p, (i,j) \in U_N, \ \ \tilde{\Delta}_i = c_i + u_i sign(i) - \sum_{p=1}^{q} \lambda^p_i r_p, i \in I^*_N, \ (13)$$

Согласно [1], совокупность множеств $K = \{U_K, I^*_K\}$ состоит из опоры $R = \{U_R, I^*_R\}$ сети G для системы (2). Опорные элементы из совокупности $R = \{U_R, I^*_R\}$ составляют сеть $G_R = (I(U_R), U_R)$, $G_R = \bigcup_{l=1}^{s} G^l_R$, которая состоит из s деревьев $G^l_R = \{I^l, U^l_R\}$, $I^l = I(U^l_R)$, $I^l_R = I^l \cap I^*_R$, $I(U_R) = I$, причём, каждое дерево содержит ровно один узел из множества I^*_R, $|I^l_R| = 1$, $l = \overline{1,s}$. Кроме элементов совокупности R в опору K входит ещё q дополнительных элементов (узлов и дуг) множества $W = \{U_W, I^*_W\}$, $|U_W| + |I^*_W| = q$, $|\Lambda_W| \ne 0$. Из (11) следует, что фундаментальными величинами критерия оптимальности являются детерминанты циклов и цепей. Укажем эффективные алгоритмы построения детерминантов, на основании которых вычисляются значения оценок

$$\tilde{\Delta}_{\tau\rho} = \Delta_{\tau\rho} - \sum_{p=1}^{q} r_p \Lambda_{\tau\rho}^{p},\ (\tau,\rho) \in U_N,\ \tilde{\Delta}_{\gamma} = \Delta_{\gamma} - \sum_{p=1}^{q} r_p \Lambda_{\gamma}^{p},\ \gamma \in I_N^{*}.$$

Элементы матриц Λ_W и Λ_N (8) также состоят из детерминантов структур [1], порожденных элементами множеств W, $N = \{U_N, I_N^{*}\}$, $U_N = U \setminus U_K$, $I_N^{*} = I^{*} \setminus I_K$. Пусть сети $G_R = \bigcup_{l=1}^{s} G_R^l$, состоящей из s деревьев $G_R^l = \left\{I^l, U_R^l\right\}$, $I^l = I(U_R^l)$, $I_R^l = I^l \bigcap I_R^{*}$, $I(U_R) = I$, причём, каждое дерево леса содержит ровно один узел из множества I_R^{*}, $\left|I_R^l\right| = 1$, $l = \overline{1,s}$ соответствует лес $T_R = \bigcup_{l=1}^{s} T_R^l$, корневых деревьев $T_R^l = \left\{I^l, D_R^l\right\}$, $I^l = I(D_R^l)$, $I_R^l = I^l \bigcap I_R^{*}$, и корнем каждого дерева леса является единственный узел из множества I_R^l, $\left|I_R^l\right| = 1$. [1, 3].

Для построения детерминантов циклов и цепей используется эффективный алгоритм [3] поиска узлов и дуг цикла (цепи) с оценкой числа операций $O(k)$, k – число узлов цикла (цепи).

В алгоритмах преобразования опоры $R = \{U_R, I_R^{*}\}$ для ввода в опору на итерациях [5] выбирается элемент (узел или дуга), для которого нарушается критерий оптимальности (11). Существуют различные правила ввода в опору приемлемой дуги (узла). Одним из известных правил выбора приемлемой дуги (узла) является выбор элемента с наибольшим нарушением критерия оптимальности (11). Основанием для такого выбора является то, что это позволяет выполнить максимальное уменьшение целевой функции на итерации. Используются также следующие альтернативы выбора элемента для ввода в опору: 1) выбор первой приемлемой дуги (узла); 2) выбор приемлемой дуги (узла) из списка кандидатов; 3) выбор приемлемой дуги (узла) по правилу острого угла [6].

Рассмотрим случай 1). Для реализации этого правила последовательно просматриваются элементы (узлы и дуги) и выбираем первый элемент, для которого нарушены условия оптимальности.

В случае 2) для строим для заданного параметра l (числа кандидатов) список узлов и дуг, для которых не выполняется критерий оптимальности (11). Затем выбираем из этого списка элемент (дугу или узел) с максимальным нарушением критерия оптимальности.

В случае 3) выбираем из оценок

$$\tilde{\Delta}_{\tau\rho} = \Delta_{\tau\rho} - \sum_{p=1}^{q} r_p \Lambda_{\tau\rho}^{p}, (\tau,\rho) \in U_N \text{ и } \tilde{\Delta}_{\gamma} = \Delta_{\gamma} - \sum_{p=1}^{q} r_p \Lambda_{\gamma}^{p}, \gamma \in I_N^*,$$

для которых нарушен критерий оптимальности (11), оценку $\bar{\tilde{\Delta}}_j$ из соотношения

$$\bar{\tilde{\Delta}}_j = \max\left\{\left|\frac{\tilde{\Delta}_k}{\sqrt{1+\sum_{i=1}^{m} x_{ik}^2}}\right|, k = 1,\cdots,n;\right\}.$$

Компонентами вектора $X_k = (x_{ik}, i = 1,\cdots,m)$ являются коэффициенты разложения k - го столбца матрицы ограничений (2) – (3) по опоре $K = \left\{U_K, I_K^*\right\}, k = 1,...,n$.

Опишем операции преобразования корневого дерева, которые необходимо выполнить в результате удаления некоторой дуги из дерева.

Пусть дереву $G_R^l = \left\{I^l, U_R^l\right\}$, соответствует корневое дерево $T_R^l = \left\{I^l, D_R^l\right\}, I^l = I(D_R^l), I_R^l = I^l \cap I_R^*$. При выполнении операции удаления некоторой дуги $(p,\ q) \in U_R^l$ из дерева G_R^l, дерево G_R^l разбивается на два поддерева: $\bar{G}_1 = (T_1,\ \bar{U}_1)$, $\bar{G}_2 = (T_2,\ \bar{U}_2)$. Обозначим через $D_1 = (T_1,\ U_1)$ и $D_2 = (T_2,\ U_2)$ корневые деревья, соответствующие деревьям $\bar{G}_1$, $\bar{G}_2$. Обозначим через $root = I_R^l$ – корень дерева T_R^l и пусть *root* принадлежит множеству T_1.

В результате удаления дуги из дерева, изменяются значения индексов уровней узлов. Однако в алгоритме удаления дуги из дерева значения индексов уровней узлов не пересчитываются. Пересчет значений индексов уровней узлов осуществляется в алгоритме добавления дуги к лесу корневых деревьев. Приведем псевдокод процедуры удаления дуги из дерева.

procedure delete_arc;

// $(p,\ q)$ – удаляемая дуга, $(p,\ q) \in U_R^l$

// Определение удаляемой дуги $(i_L,\ j_L) \in D_R^l$ корневого дерева.

if $q \neq pred[p]$ *then* $(i_L, j_L) = (p, q)$;
if $q = pred[p]$ *then* $(i_L, j_L) = (q, p)$;
// Поиск узла t
$j := i_L$;
while $thread[j] \neq j_L$ *do* $j := thread[j]$;
$t := j$;
//Поиск узла j'_L
$j := j_L$
while $depth[thread[j]] > depth[j_L]$ *do* $j := thread[j]$;
$j'_L := j$;
// Корректировка индексов предшественников и индексов связи
$pred[j_L] := 0$;
$thread[t] := thread[j'_L]$;
$thread[j'_L] := j_L$;
end;

Опишем операции преобразования корневого дерева, которые необходимо выполнить в результате добавления некоторой дуги к лесу деревьев (слияние деревьев). Пусть необходимо добавить дугу $(m,n) \in (U \setminus \bigcup^{s} U'_R) \bigcap CC(p,q)$ к лесу деревьев $\bar{G}_1 = (T_1, \bar{U}_1)$, $\bar{G}_2 = (T_2, \bar{U}_2)$, где $CC(p, q)$ – множество дуг фундаментального разреза $CC(p, q)$ [3]. В результате слияния деревьев $\bar{G}_1$ и $\bar{G}_2$ будет получено некоторое дерево $\tilde{G} = (I, \tilde{U}_0)$, $\tilde{\tilde{U}}_0 = U_0 \setminus (p, q) \bigcup (m, n)$, *или* $\tilde{U}_0 = \bar{U}_1 \bigcup \bar{U}_2 \bigcup (m,n)$. Пусть дереву $\tilde{G} = (I, \tilde{U}_0)$ соответствует корневое дерево $\tilde{D}_T = (I, \tilde{U}_T)$.

Рассмотрим алгоритм добавления дуги (i_E, j_E) к лесу корневых деревьев $D_1 = (T_1, U_1)$ и $D_2 = (T_2, U_2)$ для случая, когда $i_E \in T_1$, $j_E \in T_2$ и узел j_E является корнем дерева D_2. Заметим, что лес может состоять из нескольких деревьев. При добавлении дуги (i_E, j_E) деревья D_1 и D_2 объединяются в одно, а число деревьев уменьшается на единицу. В процедуре удаления дуги из дерева значения уровней узлов дерева D_2 не пересчитываются, поскольку уровни узлов сохраняют свои относительные значения. Однако необходимо пересчитать уровни узлов поддерева с корнем в узле j_E после добавления дуги (i_E, j_E). Очевидно, что значение уровня каждого узла поддерева с корнем в узле j_E увеличится на значение $r = depth[i_E] - depth[j_E] + 1$, где $depth[i_E]$, $depth[j_E]$ – уровни узлов корневого дерева T_R^l до удаления из него дуги (i_L, j_L). Приведем псевдокод процедуры добавления дуги (слияния деревьев):

procedure insert_arc;
// (i_E, j_E) – добавляемая дуга, j_E – корень дерева D_2
$r = depth[i_E] - depth[j_E] + 1$;
$j := j_E$;
$depth[j] := depth[j] + r$;
// определение узла j_E'
while $thread[j] \neq j_E$ *do*
begin
$j := thread[j]$;
$depth[j] := depth[j] + r$;
end;
$j_E' := j$;
// Корректировка индексов дерева
$thread[j_E'] := thread[i_E]$;
$thread[i_E] := j_E$;
$pred[j_E] := i_E$;
End.

Литература

1. Pilipchuk L.A., Malakhouskaya Y.V., Kincaid D.R., Lai M. // East-West J. of Mathematics – 2002, Vol. 4, №2. – P. 191–202.

2. Пилипчук Л.А., Пилипчук А.С. Критерий оптимальности для одной двойственной линейной экстремальной задачи // Вестник Белорусского ун-та. Сер. 1. 1998. №2. – С. 46–51.

3. Ravindra K. Ahuja, Thomas L. Magnanti, James B. Orlin. Network Flows: Theory, Algorithms, and Applica1tions. Prentice Hall, Englewood Cliffs, New Jersey, 1993. – 840 p.

4. Габасов Р., Кириллова Ф.М., Костюкова О.И. Конструктивные методы оптимизации. – Минск: Университетское, 1986. Ч. 3. – Сетевые задачи.– 222 с.

5. *Pilipchuk L.A., Pesheva Y.H., Malakhovskaya Y.V. Decomposition of* Linear Systems in Network optimization Problems // Application of Mathematics in Engineering and Economics. Proceedings of the 29 th International Summer School. Bulvest, Sofia, 2004. – P. 234 – 240.

6. Goldfarb D., Reid J. K. // Mathematical Programming. – 1977. Vol. 12, №3. – P. 361–371.

Информация об авторах:

Пилипчук Людмила Андреевна Доцент кафедры компьютерных технологий и систем факультета прикладной математики и информатики Белорусского государственного университета (БГУ) Кандидат физико-математических наук, доцент e-mail: pilipchuk@bsu.by	**Lyudmila A.Pilipchuk** Associate Professor at the Department of Computer Technologies and Systems of the Faculty of Applied Mathematics and Informatics Belarusian State University Candidate of Physico-Mathematical Sciences, Docent e-mail: pilipchuk@bsu.by
Пилипчук Андрей Степанович Соискатель кафедры методов оптимального управления факультета прикладной математики и информатики Белорусского государственного университета (БГУ) e-mail: an.pilipchuk@gmail.com	**Andrei S. Pilipchuk** Applicant at the Department of optimal control methods of the Faculty of Applied Mathematics and Informatics Belarusian State University e-mail: an.pilipchuk@gmail.com