СЕКЦІЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І АВТОМАТИЗАЦІЇ ПОЛІГРАФІЧНОГО ВИРОБНИЦТВА

УДК 655.41

І.В. Андріїв, І.В. Піх, В.М. Сеньківський

ОПТИМІЗАЦІЯ ПРОЦЕСУ ВЕРСТАННЯ СТОРІНОК КНИЖКОВИХ ВИДАНЬ З ВИКОРИСТАННЯМ ГРАФІВ

Розроблено граф зв'язків між вимогами, що відносяться до правил верстання сторінок, на основі якого здійснено оптимізацію процесу прийняття рішень, відповідних аналізу вимог до верстання елементів, взаємодії між ними та пріоритетності виконання.

Разработан граф связей между требованиями, которые относятся к правилам верстки страниц, на основе которого осуществлена оптимизация процесса принятия решений, соответствующих анализу требований к верстке элементов, взаимодействию между ними и приоритетности выполнения.

Аналіз процесу комп'ютерного верстання сторінок складних книжкових видань і вимог, що стосуються правил формування тексту та сукупного розміщення на шпальтах видання різних елементів (заголовків, таблиць, формул, виносок, ілюстрацій), а також алгоритмів його реалізації в комп'ютерних видавничих системах підтверджує наявність протиріч при так званому накладанні вимог. Це виникає тоді, коли забезпечення правил верстання одного елемента призводить до порушення інших вимог. Найчастіше такі ситуації спостерігаються на межі сторінок, тобто під час розриву тексту, яким, власне, і завершується верстання поточної сторінки й починається формування наступної.

Сформуємо деяку множину вимог $W = \{w_1, w_2, ..., w_n\}$, що стосуються правил верстання сторінок книжкових видань. Кожній з вимог присвоїмо число, або так званий ваговий коефіцієнт із множини $G = \{g_1, g_2, ..., g_n\}$, відносне значення якого відповідає важливості її дотримання. Взаємозв'язок між вимогами задамо множиною бінарних відношень B. Під бінарним відношенням B на множині W будемо розуміти другу декартову степінь множини W [2], а саме

$$B \in W \times W = W^2.$$

Виходячи із співідношення $w_i B w_j$ і враховуючи те, що $(w_i, w_j) \in W$, одержимо приналежність множини вимог до множини бінарних відношень, тобто $(w_i, w_j) \in B$. З усіх можливих вимог, які стосуються процесу формування сторінки, виберемо деяку їх підмножину, найчастіше вживану в додрукарській підготовці книжкових видань. Розмістимо вимоги в порядку зменшення ваги, тобто впливу на результат і, відповідно, на якість формування тексту та сформулюємо у вигляді певних правил:

w_1 – текст виноски і посилання на неї треба розмістити на одній сторінці;

w_2 – сторінка не повинна починатися кінцевим рядком абзаца або закінчуватися абзацним рядком (першим рядком абзаца);

w_3 – заголовки в кінці сторінки повинні бути прикриті не менш ніж трьома рядками тексту;

w_4 – при потребі можна збільшувати або зменшувати проміжки між буквами (так званий трекінґ);

w_5 – при необхідності можна збільшувати або зменшувати проміжки між рядками (так званий інтерліньяж);

w_6 – кількість суміжних рядків, які закінчуються знаком переносу, не повинна перевищувати п'яти.

Для кожної вимоги введемо вагові коефіцієнти, що ідентифікують пріоритетність її виконання і мають такі відносні значення: $g_1 = 12$; $g_2 = 10$; $g_3 = 8$; $g_4 = 6$; $g_5 = 4$; $g_6 = 2$. Сукупність вибраних вимог, що стосуються правил верстання тексту та деяких його елементів, подамо у вигляді графа, в якому множині вимог W відповідає множина точок на площині (вершини графа), а всім парам вершин (w_i, w_j) співвідносні орієнтовані лінії, що визначають дуги графа, або неорієнтовані – ребра графа.

Числові значення на вершинах графа відповідатимуть значенням вагових коефіцієнтів пріоритетності виконання вимог по забезпеченню відповідних правил верстання, значення дуг або ребер відображатимуть ступінь взаємодії вимог між собою, який можна трактувати як довжину шляху між суміжними вершинами графа.

Постановка і розв'язання задачі оптимізації з використанням графів залежить від суті і змісту задачі. У нашому випадку вона може зводитися до пошуку мінімального покриваючого дерева і шляхів з мінімальними значеннями відстаней між вершинами графа (рис.1), або, коротко кажучи, дерева мінімальних шляхів. Довжина шляху між суміжними вершинами, тобто значення ребер або дуг, що їх з'єднують, – це бінарні відношення впливу однієї вимоги на іншу. Шлях між довільними вершинами графа визначається як сума значень дуг, що входять до даного шляху.

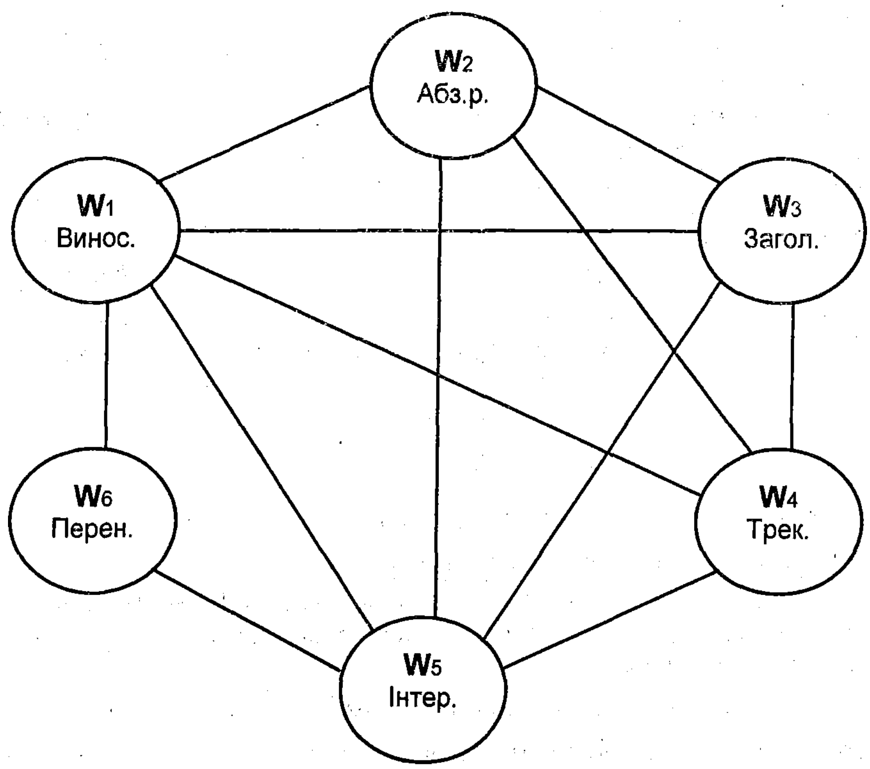

Рис.1. Граф зв'язків між вимогами до правил верстання сторінок

Побудуємо матрицю значень ребер графа, зображеного на рис.1. Кожне з цих значень вирахуємо таким чином: $p_{i,j} = (g_i + g_j)/2$ для $(i = 1,2,...,n-1; j = i+1)$. При відсутності ребра його значення вважатимемо нескінченно великим.

$$P = \begin{Vmatrix} 0 & 11 & 10 & 9 & 8 & 7 \\ 11 & 0 & 9 & 8 & 7 & \infty \\ 10 & 9 & 0 & 7 & 6 & \infty \\ 9 & 8 & 7 & 0 & 5 & \infty \\ 8 & 7 & 6 & 5 & 0 & 3 \\ 7 & \infty & \infty & \infty & 3 & 0 \end{Vmatrix}$$

Знайдемо мінімальне покриваюче дерево графа, тобто підграф, в якому сума довжин дуг, що відповідає кількості зв'язків між вершинами, мінімальна. Стосовно поставленої задачі це означатиме пошук мінімального шляху, згідно з яким треба здійснювати сукупний аналіз взаємодії вимог до правил верстання та вірогідності їх впливу на якість формування сторінки.

В орієнтованих графах шлях з вершини w_r у вершину w_s позначають як $(r-s)$-шлях [2]. У загальному випадку таких шляхів може бути декілька. Мінімальним з них буде той, сумарна протяжність складових шляхів якого найменша.

Для розв'язання задачі доволі ефективним є алгоритм Дейкстри [1], який полягає в реалізації ітеративних кроків при умові, що мінімальні шляхи до деяких k вершин визначені. Ці вершини утворюють деяку підмножину вершин $W_1 \subset W$. Далі знаходимо $(k+1)$-у вершину графа, зв'язану з вершиною W_r мінімальним шляхом. Мінімальний шлях до $(k+1)$-ї вершини повинен бути продовженням одного з раніше знайдених мінімальних шляхів. Ці можливі, з точки зору вибору, вершини утворюють підмножину $W_2 \subset W$, причому $W_1 \cap W_2$ є непустою множиною. Для всіх вершин $w_i \in W_1$ та $w_j \in W_2$ будують шляхи додаванням дуг (w_i, w_j). Відмічають вершину w_j і дугу (w_i, w_j), для якої $(r-j)$-шлях мінімальний. При $j \neq s$ шукають $(k+2)$-у вершину описаним вище способом. Під час виконання алгоритму вершинам w_j присвоюються значення q_j, які визначають мінімальні шляхи від вершини w_r до вершини w_j.

Подамо формальний опис алгоритму Дейкстри [1].

Крок 1. Відмітити вершину w_r. Прийняти $q_r = 0$, $q_j \to \infty$, $(j \neq r)$.

Крок 2. Прийняти $i = r$.

Крок 3. Для кожної невідміченої вершини j обчислити $q_j = \min\{q_j, q_i + p_{ij}\}$.

Крок 4. Перевірити умову $q_j < \infty$. (Якщо $q_j = \infty$, алгоритм закінчити, оскільки $(r-s)$-шляху у вихідному графі не існує). Відмітити ту з вершин w_j, для якої значення q_j мінімальне. Відмітити дугу на цьому кроці.

Крок 5. Прийняти $i = j$.

Крок 6. Якщо $j = s$, то $(r-s)$-шлях знайдено. Він містить відмічені дуги і є мінімальним. Якщо $j \neq s$, перейти до кроку 3.

Змінимо граф рис.1 на орієнтований, а також додатково введемо в нього вершини w_r та w_s. Згідно з умовами алгоритму одна з дуг, яка з'єднує додаткові вершини з вершинами вихідного графа, повинна мати мінімальну довжину, що дорівнює одиниці. Запишемо числові значення довжин додаткових дуг: $(w_r, w_6) = 1$, $(w_r, w_1) = 6$, $(w_3, w_s) = 2$, $(w_4, w_s) = 3$. Для застосування алгоритму Дейкстри до сформульованої вище задачі вважатимемо, що до її вершини

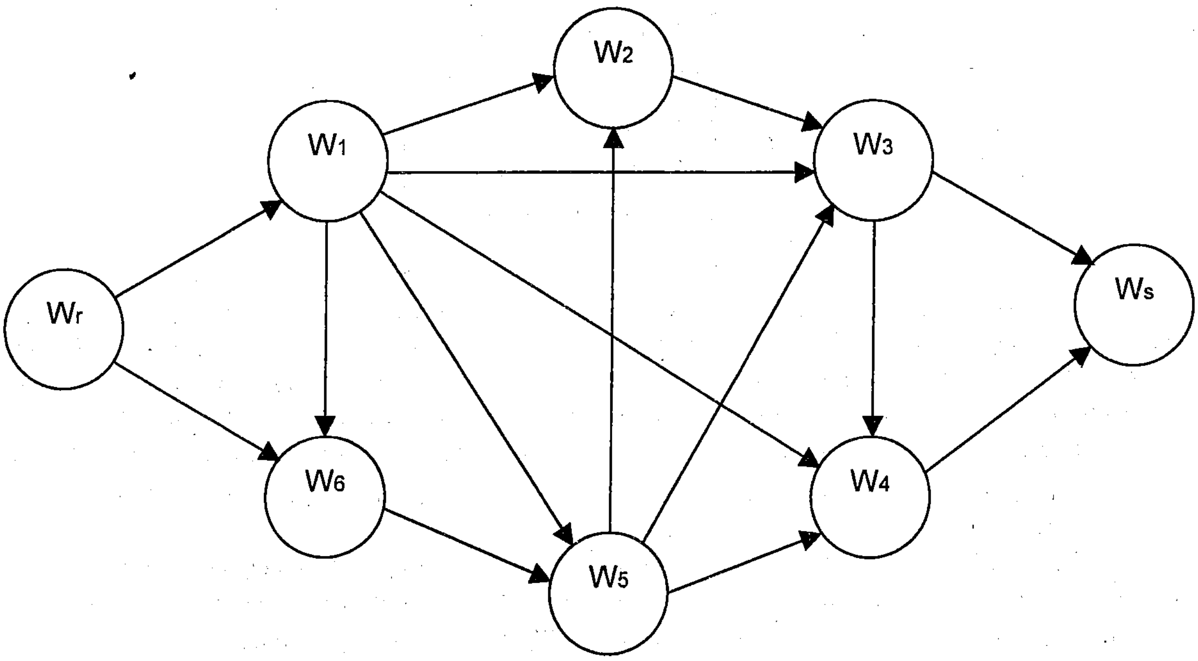

Рис. 2. Орієнтований граф для пошуку мінімального дерева

w_r мінімальний шлях знайдено. Визначимо мінімальні шляхи в графі рис.2, які ведуть до вершини w_s. При цьому наприкінці кожного етапу перевірятимемо наявність невідмічених вершин. Якщо такі вершини є, то наступний етап починається з кроку 3, інакше процес завершується.

Етап 1.

Крок 1. $q_r = 0$, $q_j = \infty$ для всіх $j \neq r$.

Крок 2. $i = r$.

Крок 3. $q_1 = \min\{q_1, q_r + p_{r,1}\} = \min\{\infty, 0+6\} = 6$,

$q_2 = \min\{q_2, q_r + p_{r,2}\} = \min\{\infty, 0+\infty\} = \infty$,

$q_3 = \min\{q_3, q_r + p_{r,3}\} = \min\{\infty, 0+\infty\} = \infty$,

$q_4 = \min\{q_4, q_r + p_{r,4}\} = \min\{\infty, 0+\infty\} = \infty$,

$q_5 = \min\{q_5, q_r + p_{r,5}\} = \min\{\infty, 0+\infty\} = \infty$,

$q_6 = \min\{q_6, q_r + p_{r,6}\} = \min\{\infty, 0+1\} = 1$,

$q_s = \min\{q_s, q_r + p_{r,s}\} = \min\{\infty, 0+\infty\} = \infty$.

Крок 4. Відмітимо вершину w_6 і дугу (w_r, w_6).

Крок 5. $i = 6$.

Крок 6. Не всі вершини відмічені.

Етап 2.

Крок 3. $q_1 = \min\{q_1, q_6 + p_{6,1}\} = \min\{6, 1+7\} = 6$,

$q_2 = \min\{q_2, q_6 + p_{6,2}\} = \min\{\infty, 1+\infty\} = \infty$,

$q_3 = \min\{q_3, q_6 + p_{6,3}\} = \min\{\infty, 1+\infty\} = \infty$,

$q_4 = \min\{q_4, q_6 + p_{6,4}\} = \min\{\infty, 1+\infty\} = \infty$,

$q_5 = \min\{q_5, q_6 + p_{6,5}\} = \min\{\infty, 1+3\} = 4$,

$q_s = \min\{q_s, q_6 + p_{6,s}\} = \min\{\infty, 1+\infty\} = \infty$.

Крок 4. Відмітимо вершину w_5 і дугу (w_6, w_5).

Крок 5. $i = 5$.

Крок 6. Не всі вершини відмічені.

<u>Етап 3.</u>

Крок 3. $q_1 = \min\{q_1,\ q_5 + p_{5,1}\} = \min\{6, 4+8\} = 6$,

$q_2 = \min\{q_2,\ q_5 + p_{5,2}\} = \min\{\infty,\ 4+7\} = 11$,

$q_3 = \min\{q_3,\ q_5 + p_{5,3}\} = \min\{\infty, 4+6\} = 10$,

$q_4 = \min\{q_4,\ q_5 + p_{5,4}\} = \min\{\infty, 4+5\} = 9$,

$q_s = \min\{q_s,\ q_5 + p_{5,s}\} = \min\{\infty, 4+\infty\} = \infty$.

Крок 4. Відмітимо вершину w_1 і дугу (w_5, w_1).

Крок 5. $i = 1$.

Крок 6. Не всі вершини відмічені.

<u>Етап 4.</u>

Крок 3. $q_2 = \min\{q_2,\ q_1 + p_{1,2}\} = \min\{11,\ 6+11\} = 11$,

$q_3 = \min\{q_3,\ q_1 + p_{1,3}\} = \min\{10, 6+10\} = 10$,

$q_4 = \min\{q_4,\ q_1 + p_{1,4}\} = \min\{9, 6+9\} = 9$,

$q_s = \min\{q_s,\ q_1 + p_{1,s}\} = \min\{\infty, 6+\infty\} = \infty$.

Крок 4. Відмітимо вершину w_4 і дугу (w_1, w_4).

Крок 5. $i = 4$.

Крок 6. Не всі вершини відмічені.

<u>Етап 5.</u>

Крок 3. $q_2 = \min\{q_2,\ q_4 + p_{4,2}\} = \min\{11,\ 9+8\} = 11$,

$q_3 = \min\{q_3,\ q_4 + p_{4,3}\} = \min\{10, 9+7\} = 10$,

$q_s = \min\{q_s,\ q_4 + p_{4,s}\} = \min\{\infty, 9+3\} = 12$.

Крок 4. Відмітимо вершину w_3 і дугу (w_4, w_3).

Крок 5. $i = 3$.

Крок 6. Не всі вершини відмічені.

<u>Етап 6.</u>

Крок 3. $q_2 = \min\{q_2,\ q_3 + p_{3,2}\} = \min\{11,\ 10+9\} = 11$,

$q_s = \min\{q_s,\ q_3 + p_{3,s}\} = \min\{12, 10+2\} = 12$.

Крок 4. Відмітимо вершину w_2 і дугу (w_3, w_2).

Крок 5. $i = 2$.

Крок 6. Усі вершини відмічені.

<u>Етап 7.</u>

Крок 3. $q_s = \min\{q_s,\ q_2 + p_{2,s}\} = \min\{12, 11+\infty\} = 12$.

Крок 4. Відмітимо вершину w_s і дугу (w_2, w_s).

Крок 5. $i = s$.

Крок 6. Усі вершини відмічені.

Побудуємо орієнтоване дерево мінімальних шляхів, яке містить відмічені вершини і дуги, що їм відповідають.

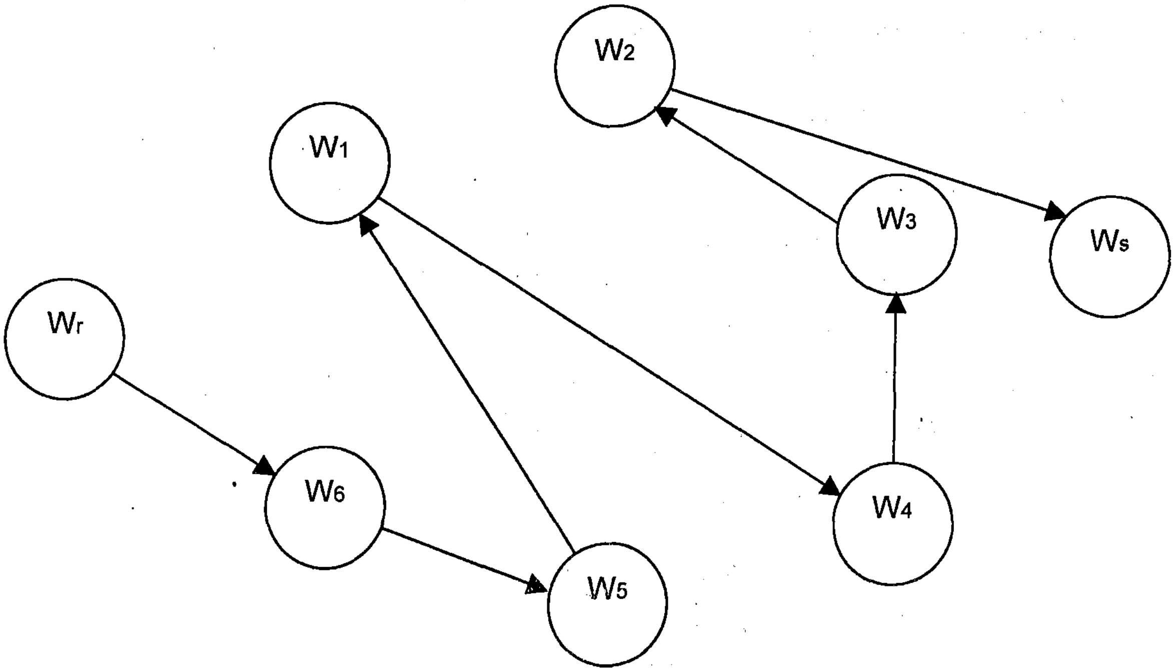

Рис.3. Орієнтоване дерево мінімальних шляхів між правилами верстання сторінок

Отже, якщо в процесі додрукарської підготовки видання маємо сукупність перерахованих вище вимог до правил формування тексту й верстання його складових частин, то мінімальна протяжність структурних зв'язків і взаємодія між вимогами описується орієнтованим деревом (рис.3), побудованим на основі аналізу зваженого графа та певних операцій над значеннями його вершин (ваговими коефіцієнтами пріоритетності виконання вимог) і дуг (числовими векторами).

Запропонований метод дозволяє визначити мінімальні шляхи переходів від результатів виконання однієї вимоги до другої. Зрозуміло, що при наявності у виданні інших елементів складності граф взаємозв'язків між ваговими коефіцієнтами пріоритетності виконання вимог, які відповідають правилам верстання цих елементів, та орієнтоване дерево мінімальних шляхів будуть видозмінені.

1. Куликовский Л.Ф., Мотов В.В. Теоретические основы информационных процессов. М., 1987. 2. Майника Э. Алгоритмы оптимизации на сетях и графах: Пер. с англ. М., 1981. 3. Пономаренко О.І., Пономаренко В.О. Системні методи в економіці, менеджменті та бізнесі. К., 1995. 4. Хэмминг Р.В. Теория кодирования и теория информации. М., 1983.

УДК 621.373.826

О.В. Ющик

ОСОБЛИВОСТІ ДОДРУКАРСЬКОЇ ТЕХНОЛОГІЇ "КОМП'ЮТЕР–ДРУКАРСЬКА ФОРМА"

Висвітлюються технологічні особливості впровадження у виробництво додрукарської технології "комп'ютер–друкарська форма".

Освещаются технологические особенности производственного внедрения допечатной технологии "компьютер–печатная форма".