

ООО «Институт креативных технологий»

**ГАРМОНИЧНОЕ РАЗВИТИЕ
СИСТЕМ – ТРЕТИЙ ПУТЬ ЧЕЛОВЕЧЕСТВА**

Коллективная монография
по материалам трудов
1-го Международного Конгресса
Одесса, 8 – 10 октября 2011 г.

под ред. Сороко Э.М., Егоровой-Гудковой Т.И.

ООО «Институт креативных технологий»
Одесса – 2011

ББК 65. 012
УДК 330.131.7

Рекомендовано к публикации Ученым советом
Института математики, экономики, механики Одесского Национального
университета им. И.И. Мечникова
протокол № 5 от 4 июля 2011 г.

Сороко Э.М., Егорова-Гудкова Т.И.
ГАРМОНИЧНОЕ РАЗВИТИЕ СИСТЕМ – ТРЕТИЙ ПУТЬ
ЧЕЛОВЕЧЕСТВА
Коллективная монография
Одесса, ООО «Институт креативных технологий», 2011, 395 стр.

Коллективная монография сформирована на основе научных трудов и докладов участников Конгресса «ГАРМОНИЧНОЕ РАЗВИТИЕ СИСТЕМ – ТРЕТИЙ ПУТЬ ЧЕЛОВЕЧЕСТВА».

Книга предназначена для учёных, специалистов, аспирантов и всех, кого интересует вопрос, как выжить Миру?, в чём суть Третьего пути Человечества и каков он этот Новый путь.

ISBN 978-966- 8888- 03-9

© ООО «Институт креативных технологий», 2011

АДАПТИВНИЙ МЕТОД УЩІЛЬНЕННЯ ДАНИХ НА ОСНОВІ ЛІНІЙНОЇ ФОРМИ ФІБОНАЧЧІ

*Об авторях: Лужецький В. А., д.т.н., професор, Савицька Л.А., асистент
Вінницький національний технічний університет, м. Вінниця, Україна*

Вступ

Існуючі методи й алгоритми ущільнення даних, в основному, враховують статистику символів у повідомленні [1-3] або базуються на побудові словника [4,5]. І хоча на практиці широко використовуються архіватори, створені на основі саме цих методів, пошуки нових підходів до ущільнення даних продовжуються.

Одним із цікавих підходів є пропозиція використовувати для ущільнення даних оптимізуючі властивості чисел Фібоначчі [6,7]. Суть підходу полягає в тому, що в процесі ущільнення інформації блок цифрових даних будь-якої довжини розглядається як надвелике ціле додатне число, що представляється набором із трьох невеликих чисел. Таке представлення чисел, називається лінійною формою Фібоначчі.

Теоретичні дослідження і практика застосування архіваторів показали, що не існує універсального неадаптивного методу ущільнення, який забезпечував би однаковий коефіцієнт ущільнення для різних типів даних. Тому наукові дослідження найчастіше спрямовані на створення ефективних методів ущільнення певних типів даних. Однак дані навіть одного типу, з погляду ущільнення, мають різні властивості і характеристики. З огляду на це, останнім часом прагнуть до створення адаптивних алгоритмів ущільнення даних [8,9]. Запропоновані в роботах [6,7] методи ущільнення даних на основі лінійної форми Фібоначчі є неадаптивними і тому не можуть забезпечити однаково високий коефіцієнт ущільнення для різних типів даних.

Мета даної роботи – показати можливість підвищення ефективності ущільнення даних на основі лінійної форми Фібоначчі за рахунок адаптації виконуваних перетворень до конкретного змісту ущільнюваних даних.

Постановка задач

Одним з найважливіших положень теорії ущільнення інформації є висловлена в [10] ідея поділу процесу ущільнення на дві процедури: моделювання і кодування. Моделювання визначає характеристики джерела даних, що ущільнюються, а кодування перетворює символи у послідовність бітів відповідно до отриманих характеристик.

У роботі [11] запропоновано таку узагальнену модель процесу адаптивного ущільнення:

$$C_A = \{P, A, M, C, D, P_M, P_C, P^*, S, f\}$$

де P - вихідна послідовність символів алфавіту $A = \{0,1\}$;

$M = \{M_i\}$ - множина правил моделювання джерела даних;

$C = \{C_j\}$ - множина правил кодування даних;

$D = \{D_j\}$ - множина правил декодування даних;

$P_M = \{P_{M_i}\}$ - множина послідовностей, що є результатом моделювання;

$P_C = \{P_{C_j}\}$ - множина послідовностей, що є результатом кодування;

P^* - послідовність ущільнених даних;

S - правило формування структури послідовності P^* ;

f - функція оптимізації.

Саме наявність множини правил моделювання джерела даних і множини правил кодування даних забезпечують можливість адаптації до конкретного змісту ущільнюваних

перетворення в лінійну форму Фібоначчі.

Таким чином, маємо чотири правила кодування $C = \{C_0(0,0), C_1(0,1), C_2(1,0), C_3(1,1)\}$, які породжують відповідні структури кодованих блоків (табл.1). Причому кожне із правил передбачає формування ознак структури p_1 і p_2 . Тут l_{a_0} - розрядність коду числа a_0 ; l_{b_0} - розрядність коду числа b_0 .

Таблиця 1 Правила кодування і структури блоків

Позначення правила кодування	Ознаки		Позначення структури блоку	Вигляд структури блоку
	p_2	p_1		
$C_0(0,0)$	0	0	STR0	$\begin{matrix} x \times K \times \ 0 \ 0 \\ 1 \ 2 \ 3 \\ \hline \end{matrix}$
$C_1(0,1)$	0	1	STR1	$\begin{matrix} 1x \times K \times \ q \ 0 \ 1 \\ 14 \ 24 \ 3 \ 1 \ 2 \ 3 \\ \hline \end{matrix}$
$C_2(1,0)$	1	0	STR2	$b_0 \ a_0 \ j \ l_{b_0} \ l_{a_0} \ 1 \ 0$
$C_3(1,1)$	1	1	STR3	$b_0 \ a_0 \ j \ l_{b_0} \ l_{a_0} \ 1 \ 1$

Оскільки блоки перетвореної послідовності даних, у загальному випадку, будуть мати різну структуру, то результатом ущільнення є неоднорідна послідовність. Виходячи із цього, функція оптимізації, що описує правило вибору однієї з чотирьох структур перетвореного блоку, має вигляд:

$$f_{\text{opt}}^{\text{min}} = \min\{l_0, l_1, l_2, l_3\},$$

де l_0, l_1, l_2, l_3 - довжина блоку структури STR0, STR1, STR2, STR3, відповідно.

У загальному випадку, при розбитті вихідних даних на блоки довжини 2^a один блок може мати довжину менше 2^a . Тому необхідно або вказати довжину вихідних даних L і потім обчислити довжину меншого блоку, або вказати довжину l_K цього блоку. Другий варіант вимагає меншої розрядності коду, тому вибираємо його. Таким чином, додаткова інформація містить у собі значення розрядності 2^a і l_K .

Отже маємо таке правило формування структури послідовності P^* , яке представляється у вигляді кортежу:

$$S = \{ \text{Бл}^* K \| K \text{Бл}^* 2 \| \text{Бл}^* 1 \| l_K \| 2^a \}.$$

Розрядність кодованих блоків $\text{Бл}^* i$ ($i=1+K$) різна. Розрядність вихідних блоків може досягати значення 2^{16} , тому розрядність полів l_K і 2^a дорівнює 16.

З урахуванням вищесказаного, реалізація адаптивного методу ущільнення даних одним проходом передбачає виконання таких обчислень і перетворень.

Спочатку вводиться послідовність даних P , що підлягають ущільненню, і визначається довжина L цієї послідовності. Потім вводяться значення параметру a і обчислюється кількість блоків $K = \lfloor L/2^a \rfloor$, на які буде розбиватися послідовність P . Потім обчислюється довжина K -го блоку $l_K = L - (K-1)2^a$. Після цього створюється початкова послідовність ущільнених даних P^* , що складається із двох елементів: l_K і 2^a . Усі описані дії стосуються ініціалізації процесу ущільнення.

Безпосередньо процес ущільнення складається з K циклів. У кожному з них зчитується черговий блок вихідних даних $\text{Бл}^* i$, над яким виконуються перетворення згідно правил кодування $C_0(0,0)$, $C_1(0,1)$, $C_2(1,0)$, $C_3(1,1)$ і створюються структури STR0, STR1, STR2, STR3. Для кожної структури визначається її довжина та відповідно до правила оптимізації

$\min\{l_0, l_1, l_2, l_3\}$ вибирається структура найменшої довжини, якою доповнюється послідовність ущільнених даних P^* .

Процес відновлення даних визначається структурою послідовності ущільнених даних

$$S = \{ \text{Бл}^* K \parallel K \text{Бл}^* 2 \parallel \text{Бл}^* 1 \parallel l_K \parallel 2^a \}$$

і множиною правил декодування

$$D = \{ D_0(0,0), D_1(0,1), D_2(1,0), D_3(1,1) \}.$$

Розглянемо правила декодування даних для кожної з описаних вище структур.

Вибір правила декодування для заданого блоку здійснюється на основі аналізу ознак p_1 і p_2 структури блоку. Після вибору правила декодування розряди ознак відкидаються і подальші дії виконуються над основною частиною структури блоку.

Якщо $p_1=0$ і $p_2=0$, то це означає, що блок має структуру STR0, до якої застосовується правило декодування $D_0(0,0)$. Дане правило полягає в тому, що основна частина структури блоку залишається незмінною.

Якщо $p_1=1$ і $p_2=0$, то до блоку структури STR1 необхідно застосувати правило $D_1(0,1)$. Воно полягає у виконанні таких дій. В основній частині структури

$$\begin{array}{cccc} 1x & x & K & x \\ 14 & 2 & 43 & 1 \\ & & 2^{-q} & 2^a \end{array} \parallel q$$

відкинути a молодших розрядів і до частини, що залишилася, дописати q старших нулів. У результаті цього буде отримано блок вигляду:

$$\begin{array}{cccc} 00K & 01x & x & K & x \\ 1 & 2 & 3 & & \\ 1 & 4 & 42 & 4 & 43 \\ & & 2^m & & \end{array}$$

Якщо $p_1=0$ і $p_2=1$, то до блоку структури STR2 застосовується правило $D_2(1,0)$. За цим правилом, виходячи з чисел j , a_0 і b_0 , виконується зворотне перетворення Фібоначчі, тобто обчислюється за формулою (1) ціле додатне число N , що відповідає даній лінійній формі Фібоначчі. Після цього визначається розрядність l^* двійкового коду числа N і обчислюється різниця $\Delta = 2^a - l^*$. Дана різниця показує, скільки нулів необхідно дописати в старші розряди коду числа N .

Якщо $p_1=1$ і $p_2=1$, то до блоку структури STR3 застосовується правило $D_3(1,1)$. Воно передбачає обчислення числа за заданою лінійною формою Фібоначчі й заміну самої старшої одиниці в коді цього числа на нуль.

Відновлення вихідних даних з ущільнених передбачає виконання таких обчислень та перетворень.

Спочатку вводиться послідовність даних P^* , які підлягають відновленню. Потім зчитуються 16 наймолодших розрядів, що є кодом числа 2^a . Після цього дані розряди видаляються з послідовності P^* . На наступному кроці також зчитуються 16 наймолодших розрядів, що є кодом числа l_K . Далі створюється порожня послідовність P . Усі ці дії спрямовані на ініціалізацію процесу відновлення.

Безпосередньо процес відновлення складається з K циклів. На початку кожного циклу зчитується два наймолодші розряди послідовності P^* , що є кодом ознаки структури. Відповідно до цього коду реалізується певне правило декодування. Результатом є черговий блок відновлених даних, що додається в послідовність P . Структура, з якої отриманий цей блок, видаляється з послідовності P^* .

Відомі [6,7] методи ущільнення даних на основі лінійної форми Фібоначчі передбачають тільки одне правило кодування $C_2(1,0)$ для кожного блоку вхідних даних. При цьому можливі такі випадки:

- 1) структура STR2 перетвореного блоку має довжину, що дорівнює довжині вихідного блоку;
- 2) структура STR2 перетвореного блоку має довжину, що менша за довжину вихідного блоку;
- 3) структура STR2 перетвореного блоку має довжину, що більша за довжину вихідного блоку.

У першому випадку немає ефекту ущільнення, у другому – є, а у третьому випадку відбувається збільшення розрядності даних. Виходячи з цього коефіцієнт ущільнення даних визначається за формулою:

$$S_{\text{ущ}}^{\text{на}} = \frac{L}{w_1 2^a + l_2 + l_3},$$

де $l_2 = \sum_{i=1}^{w_2} l_i$ - сумарна довжина перетворених блоків, що відповідають випадку 2;

$l_3 = \sum_{j=1}^{w_3} l_j$ - сумарна довжина перетворених блоків, що відповідають випадку 3;

w_1, w_2, w_3 - кількість перетворених блоків, що відповідають випадкам 1, 2 і 3;

$$w_1 + w_2 + w_3 = K.$$

У разі застосування запропонованого адаптивного методу ущільнення випадок 3 виключається, а замість нього буде випадок 1. При цьому коефіцієнт ущільнення даних буде визначатися за формулою:

$$S_{\text{ущ}}^{\text{а}} = \frac{L}{w_1 2^a + l_2},$$

де $w_1 = K - w_2$.

Експериментальні дослідження запропонованого методу ущільнення для файлів різного змісту та розмірів показали, що коефіцієнт ущільнення збільшується від 2 до 6 разів порівняно з неадаптивним методом ущільнення, описаним в роботі [7].

Висновки

Основна перевага узагальненої моделі адаптивного процесу ущільнення даних полягає в тому, що формалізується підхід щодо розробки нових методів ущільнення. Це показано на прикладі формулювання нових правил моделювання джерела даних, правил кодування даних на основі лінійної форми Фібоначчі та опису функцій оптимізації.

Подальше підвищення ефективності ущільнення даних автори бачать в адаптації до розрядності блоків, на які розбиваються дані, та кількості послідовних процедур ущільнення даних. Остання можливість забезпечується саме завдяки використанню лінійної форми Фібоначчі.

Список літератури

1. Huffman D.A. A method for the construction of minimum redundancy codes // Proceedings of the Institute of Electrical and Radio Engineers. – 1952. – V. 40, 9. – P. 1098-1101.
2. Cormack G.V., Horspool R.N. Data compression using dynamic Markov modeling // Comput. J. – 1987. – V. 30, 6. – P. 541-550.
3. Storer J. A. Data Compression: Methods and Theory. //Computes Science Press. - 1988. - V. 47, 1. – P. 23-29.
4. Ziv J., Lempel A. Compression of individual sequences via variable-rate coding // IEEE Trans. Inf. Theory. – 1978. – V. IT-24, 5. – P. 530-536.
5. Welch T.A. A Technique for High Performance Data Compression. // Computer/ - 1984. - №6. - P. 176-189.

6. Анисимов А. В., Рындин Я. П., Редько С. Е. Обратное преобразование Фибоначчи // Кибернетика. -- 1982. - № 3. - С. 9-11.
7. Кшановський О.Д., Титарчук С.В., Лужецький В.А. Арифметичні методи ущільнення цифрової інформації // Вісник ВПІ. - 1999. - №5. - С. 83-87.
8. Кричевский Р.Е. Сжатие и поиск информации. - М.: Радио и Связь, 1989. - 176 с.
9. Рябко Б.Я., Фионова А.Н. Эффективный метод адаптивного арифметического кодирования для источников с большими алфавитами. // Проблемы передачи информации. - 1999. - Том 35, вып. 4. - С.34-39.
10. Rissanen J.J., Langdon G.G. Universal modeling and coding // IEEE Trans. Inf. Theory. - 1981. - V. IT-27, 1. - P. 12-23.
11. Лужецький В. А., Савицька Л. А., Шахзада Ашрафул Хок. Узагальнена модель адаптивного ущільнення даних // Інформаційні технології та комп'ютерна інженерія. - 2010. - №2. - С. 54-59.