

Я.Ю. Жураковський, аспір.

С.М. Запольський, студ.

Національний технічний університет України "КПІ"

ПОРІВНЯЛЬНА ОЦІНКА СПОСОБІВ СТИСНЕННЯ ВИМІРЮВАЛЬНОЇ ІНФОРМАЦІЇ

Виконується аналіз та наводиться порівняльна оцінка деяких методів стиснення вимірювальної інформації, які використовуються в автоматизованих системах керування технологічними процесами (АСК ТП) для попередньої обробки інформації, що надходить від первинних датчиків. Аналіз та порівняльна оцінка способів стиснення виконані за допомогою запропонованого в роботі алгоритму проведення досліджень, який дозволяє одержати дані про основні показники ефективності стиснення: коефіцієнт та швидкість. Наводяться рекомендації щодо використання розглянутих методів стиснення в АСК ТП.

Вимірювальна інформація, що надходить від датчиків до пунктів збору первинної інформації в автоматизованих системах керування технологічними процесами (АСК ТП), як правило, має значну надмірність. Для зменшення навантаження на тракти передачі АСК ТП виконують стиснення вимірювальної інформації, для чого використовують такі методи стиснення, як: LZH, LZW, Vitter, матричний та каскадні [1, 2].

З метою проведення досліджень, для аналізу кожного з методів стиснення відносно вимірювальної інформації та розробки рекомендацій з їх застосування в АСУ ТП були запропоновані алгоритм та схема досліджень методів стиснення.

В якості основних показників ефективності стиснення при проведенні досліджень обрали: коефіцієнт та швидкість стиснення. Виміри проводилися окремо для кожного з параметрів, при цьому досліджувалась залежність показників ефективності від розміру переданого блоку даних.

На рис. 1 наведена схема проведення досліджень. Генератор тестових послідовностей видає фрагмент вимірювальної інформації. Цей фрагмент надходить у модуль обробки і відображення, що здійснює виклик зазначених процедур стиснення різними методами й одержує від них стиснуті послідовності. При вимірі швидкості стиснення він також виконує запуск таймера. Маючи довжину первинного (вхідного) і стиснутого (вихідного) фрагментів, основний вимірювальний модуль обчислює коефіцієнт стиснення, та, порівнюючи показання таймера перед викликом процедури комресії і після нього, одержує час стиснення. Потім результати вимірювань надходять на модуль відображення даних, що представляє їх у вигляді графіків.

Для того, щоб щонайкраще оцінити ефективність алгоритмів стиснення, в якості вхідних даних необхідно використовувати фрагменти, які за своїми статистичними властивостями були б максимально наближеними до реальної інформації. Найкраще використовувати фрагменти даних, що отримані від реальних джерел, при цьому всю інформацію необхідно представити в уніфікованому вигляді. Тому в якості вхідних даних для кодера використані інформаційні фрагменти вимірювальної інформації, що надходить від датчиків (температури, тиску, витрати електроенергії тощо).

Для зручності кодування алфавіт вхідного і вихідного кодів обраний таким, що дорівнює 256 символам (для кодування одного символу використовується один байт). У тих випадках, коли вихідний код вимагає його розширення, на вході використовується алфавіт із 128 символів (використовуються молодші 7 біт кодової комбінації).

Для порівняльних досліджень були вибрані такі методи стиснення: LZH, LZW, Vitter, матричний та каскадні [3, 4].

При кодуванні методами LZH, LZW розмір блока даних, що передається, впливає на швидкість передачі, а особливо на коефіцієнт стиснення. Це характеризується тим, що для цих методів складається кодова таблиця, в яку записуються словосполучення, що надходять від вхідного потоку, і надалі розглядається як префікс рядка. Кожному префіксу привласнюється свій номер, що передається в канал. Чим більше елементів входить у кодову таблицю, тим більше коефіцієнт стиснення. Але пошук номера, що відповідає префіксу, впливає на затримку інформаційного потоку, що є важливим для систем передачі інформації, які працюють у режимі реального часу. Для подолання вищказаної затримки в LZW алгоритмах застосовують хешування.

При кодуванні методом Vitter відбувається динамічна побудова хашфменовського дерева. Даний метод дозволяє визначити положення батьківських і дочірніх вузлів дерева за фіксова-

ний час шляхом обчислення зсуву від показчиків батька та його правого дочірнього вузла лідера блока. Це дозволяє вузлу ковзати по блоку, модифікуючи постійне число показчиків. Виконання процедури при цьому займає певний постійний час.

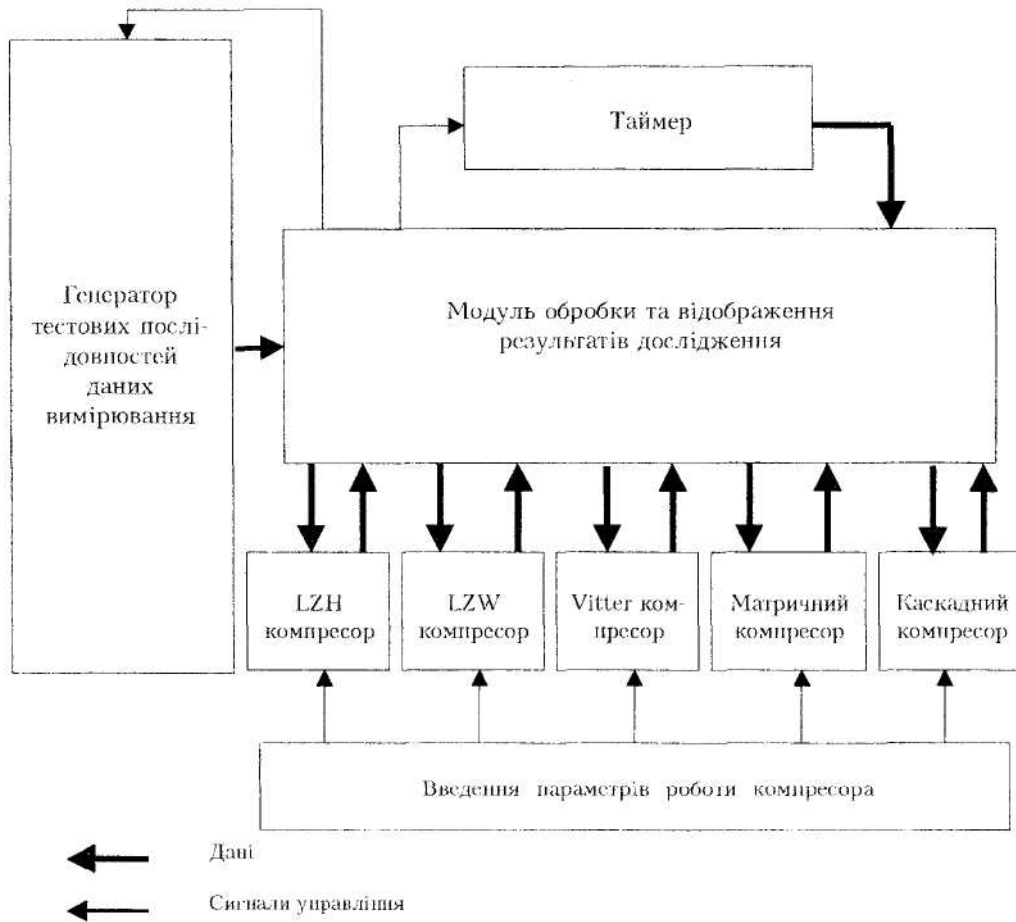


Рис. 1

При кодуванні матричним методом стиснення здійснюється в два проходи, тому затримка даних на виході кодера пропорційна розміру блока даних, що передається. Швидкість стиснення залежить від розміру блока: чим більший блок даних, тим більшу кількість матриць потрібно проаналізувати. Коефіцієнт стиснення зростає зі збільшенням розміру блока, а також залежить від характеру вхідного інформаційного потоку.

Виходячи з цього, основною задачею досліджень є відпукання найбільш оптимального співвідношення параметрів алгоритму стиснення для різних вхідних потоків інформації.

Алгоритм основної процедури досліджень

Основна процедура досліджень є оболонкою, що формує вхідні дані для процедур стиснення (компресії), виконує їх запуск, визначає параметри методу стиснення, що досліджується, та представляє їх у зручному для користувача вигляді.

Алгоритм основної процедури досліджень показаний на рис. 2. Для роботи алгоритму необхідно задати метод стиснення, що досліджується, параметри стиснення, а також вибрати вхідний потік, на якому буде робитися тестування. Один із параметрів стиснення вказується як такий, що варіюється. Обираємо таким параметром блок вхідної інформації, що змінюється в межах від 4 до 4 096 байт. У процесі вимірювання цей параметр змінюється в зазначених межах, а результати для кожного значення записуються у вихідний масив.

При визначенні швидкості стиснення процедура стиснення виконується в циклі визначене число разів, щоб підвищити точність вимірювань мінімізувати вплив сторонніх процесів (переривання, черкнення задач та ін.) на результат вимірів.

Після проведення вимірювань, за їх результатами робиться обчислення показників стиснення. Ці показники систематизуються і видаються користувачеві у вигляді графіків.

Алгоритми стиснення

Процедура динамічного стиснення **LZW**-методом починається з заповнення **LZW**-таблиці. До неї вносяться рядки, що відображають статистику повідомлень на визначеній мові. При відсутності такої статистики таблиця містить тільки рядки, що складаються з одиночних символів. Потім, у міру надходження даних від джерела інформації, формуються рядки, що складаються з декількох символів. Виходячи з того, що таблиця має обмежений розмір, рядки, що зустрічаються в тексті не часто, вилучаються, а на їхнє місце вносяться рядки, що мають більшу частоту появи. Таким чином, у процесі накопичення статистики про повідомлення, що стискається, відбувається динамічна перебудова таблиці кодування й адаптація її до характеру переданих даних.

Метод стиснення **LZH** ґрунтується на використанні двох методів стиснення даних: словниково-му (**LZ**) і статистичному (**Huffman**). На початку виконання алгоритму стиснення відбувається ініціалізація словника. Він логічно являє собою сукупність абстрактних структур даних, що містять набір деревоподібних структур, в яких кожний корінь відповідає певному знаку алфавіту. При 8-розрядному форматі символу кількість таких дерев дорівнює 256. Деревоподібні структури являють собою набір відомих рядків, що починаються одним визначеним символом, а кожний вузол дерева представляє один рядок із цього набору. При виконанні стиснення циклічно формується новий рядок шляхом додавання чергового одиночного символу до існуючого рядка, який знаходиться в таблиці статей словника, що виражається в доповненні новим вузлом кодового дерева.

Далі відбувається зіставлення рядка, при якому послідовність символів із потоку даних зіставляється зі статтею словника. Якщо рядок відповідає статті словника і не є створеним під час останнього виклику процедури зіставлення рядка, то вводиться наступний символ і додається до рядка. Потім цей крок повторюється. Якщо рядок не відповідає статті словника, або відповідає статті, яка створена під час останнього виклику процедури зіставлення рядків, то останній символ відкидається, а кодове слово, що відповідає цій статті словника, подається на вихід кодера.

Головною особливістю алгоритму **Vitter**, є те, що дерево будується за принципом ковзання і збільшення, використовуючи при цьому умову неявної нумерації. Суть цієї операції полягає в тому, що вузол, оголошений поточним, обміняється з лідером свого блоку і потім ковзає в напрямку кореня дерева по сусідньому блоку, який безпосередньо прилягає до блоку поточного вузла. Ковзання продовжується доти, поки поточний вузол не пройде весь блок і не буде встановлений на початку цього блоку. Потім здійснюється збільшення ваги поточного вузла і новим поточним вузлом призначається батько старого поточного вузла. Операція ковзання зі збільшенням продовжується до досягнення кореня дерева.

Розглянута структура дерева називається "плаваючим деревом" тому, що покажчики батьківського та його дочірніх вузлів підтримуються неявно. Кожний блок має лише покажчики батьківських і правих дочірніх вузлів лідера блоку. Через безперервність пам'яті, де зберігаються зовнішні і внутрішні вузли, положення батьківських і дочірніх вузлів дерева, інших вузлів блока можуть бути визначені за фіксований час шляхом обчислення зсуву від покажчиків батька і його правого дочірнього вузла лідера блоку. Це дозволяє вузлу ковзати по блоку, модифікуючи постійне число покажчиків. При цьому час виконання процедури залишається постійним, що збільшує швидкість кодування при значних масивах інформації.

Для роботи алгоритму **матричного** стиснення необхідно задати розмір блока даних. Для зручності та наочності вхідний потік розбивається на рядки символами "переведення рядка – повернення каретки". У реальних умовах довжина рядка матриці визначається довжиною запису даних, що передаються. Інформація зчитується блоками з вхідного файлу, одночасно робиться її аналіз і пошук елементів, що повторюються, у сусідніх рядках. Якщо в двох сусідніх рядках знайдені однакові ланцюжки символів, алгоритм здійснює перегляд наступних рядків, визначаючи кількість рядків матриці. Розміри матриці і помери рядків для кожного рядка матриці записуються в дошоміжний масив. Якщо розміри матриці задовольняють визначеній умові, то вона позначається як та, що підлягає стисненню.

При записі інформації у вихідний потік для кожної виділеної матриці передається її перший рядок, перед яким ідуть ознака стиснення і розміри матриці. Наступні рядки матриці опускаються.

Каскадний метод поєднує в собі стиснення інформації двома методами. Його відмінність від інших методів, поданих у роботі, в тому, що дані спочатку стискаються одним методом, а потім отриманий стиснутий масив обробляється іншим методом. Розгортання інформації робиться в зворотному порядку. У цій роботі використовуються чотири види алгоритмів **LZH**, **LZW**, **Vitter** та матричний. У програмному забезпеченні, яке було розроблено при виконанні цієї роботи, розглянуті усі види комбінацій поданих алгоритмів. При цьому кількість можливих варіантів складає 12, тому що розглядалися тільки парні сполучення алгоритмів.

При реалізації каскадного методу стиснення може виникнути проблема неоднозначності трактування стиснутого масиву даних. Це може відбутися в тому випадку, якщо алгоритм стиснення першого каскаду в результаті роботи згенерує масив даних, що містить керуючі символи алгоритму другого каскаду. При розгортанні (декомпресії) такого масиву процедура другого каскаду інтерпретує ці символи, як ознаки стиснення, і інформація буде спотворена. Ця проблема може бути вирішена двома способами. Перший полягає в тому, що при стисненні робиться розширення алфавіту джерела, при цьому алфавіт для процедури стиснення другого каскаду складається з вихідних і керуючих символів першого каскаду. Другий спосіб передбачає чіткий поділ способу вказівки ознаки стиснення для кожного каскаду, що не дозволяє використання однакових символів. Наведений на рис. 2 алгоритм каскадного методу стиснення реалізує другий спосіб.

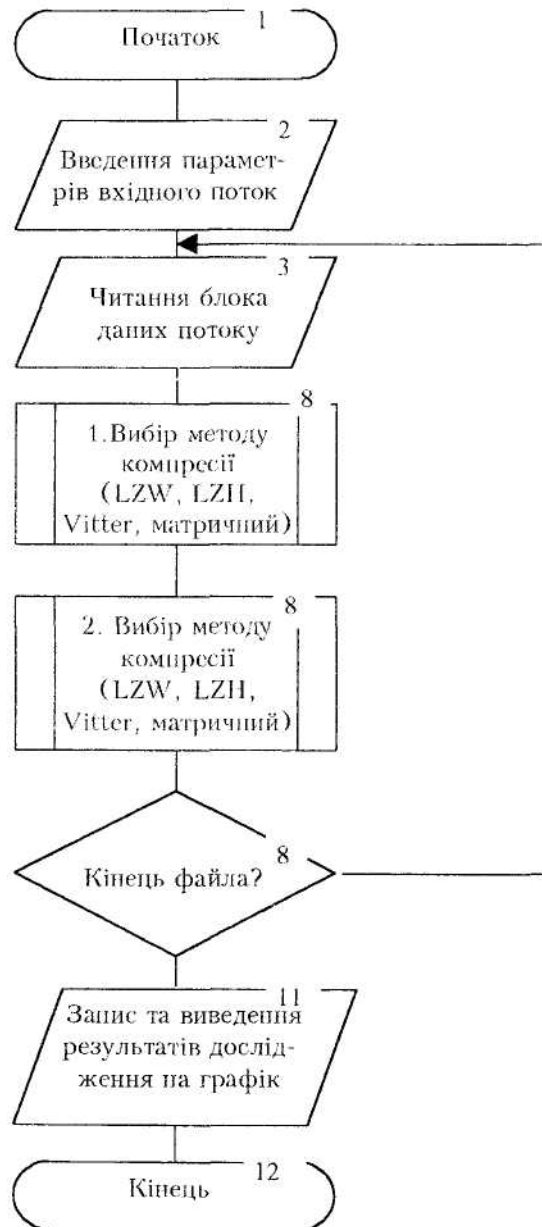


Рис. 2

Програмна реалізація алгоритмів стиснення

Всі процедури стиснення в даній роботі написані на мові Pascal із застосуванням ключів, що забезпечують оптимальність алгоритму, а також незалежність швидкості роботи скомпільованої програми від транслятора та найкращий контроль над ресурсами процесора. Написані на мові Pascal, процедури дозволяють проводити дослідження на різних процесорах – від 18 086 до Pentium. Під

час вимірювання всі дані, які необхідні для роботи, знаходяться в ОЗП, що виключає вплив швидкості і кеша жорсткого диска на результат вимірів. Щоб мінімізувати вплив сторонніх процесів, дослідження відбувається при заборонених зовнішніх перериваннях, заборонено також переключення задач.

Коефіцієнт стиснення визначається як відношення довжини первинного масиву інформації до довжини стиснутого масиву. Затримка даних визначається програмою під час роботи як максимальна різниця між позиціями показників первинного та стиснутого масивів. Для вимірювання швидкості роботи алгоритму використовується внутрішній таймер ПК. Оскільки збільшення значення таймера відбуваються приблизно 18 разів у секунду, що співвідноситься з часом роботи алгоритму, то для підвищення точності вимірювання робиться багатократне виконання процедури стиснення.

Для наочного візуального відображення отриманих результатів була створена оболонка, що дозволяє здійснити запуск процедур компресії та відобразити потім результати дослідження у вигляді графіків. Передбачено також можливість проведення досліджень на різних системах і спільному опрацюванні результатів на одній машині. Оболонка написана на мові Object Pascal у системі програмування Borland Delphi 4.0 Client/Server для операційної системи Windows 95/98/Windows NT 4.0.

Оцінка коефіцієнта стиснення

Мета такого дослідження – визначити для вимірювальної інформації найбільш оптимальний метод стиснення, встановити залежність коефіцієнта стиснення від довжини блока даних та на підставі цієї залежності визначити оптимальну довжину блока.

При порівнянні ефективності стиснення для розглянутих методів використовувався коефіцієнт K_c , що показує, у скільки разів зменшився обсяг повідомлення після стиснення стосовно первинного (вхідного) обсягу.

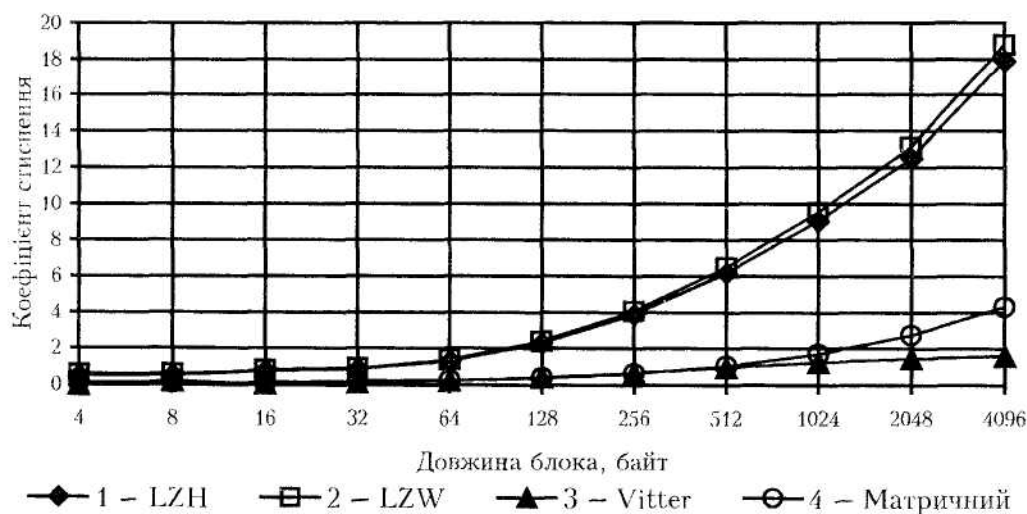


Рис. 3

На рис. 3 показана залежність коефіцієнта стиснення від обсягу первинної вимірювальної інформації при стисненні різними методами, де крива 1 відповідає стисненню методом LZH, 2 – LZW, 3 – Vitter, 4 – матричним; на рис. 4 – при стисненні каскадними методами, де крива 1 відповідає стисненню методом LZW-Vitter, 2 – Vitter-LZH, 3 – матричний-Vitter, 4 – матричний-LZW.

Як видно з рисунка 3, для вимірювального інформаційного потоку істотне збільшення коефіцієнта стиснення для методу LZH (крива 1) починається з блока 128 байт. Це пояснюється тим, що відбувається набір кодової таблиці, і тим, що цей метод є префіксним. Тому, чим більше таблиця містить у собі елементів, тим більша ймовірність збігу елементів таблиці з вхідним потоком, а це призводить до збільшення коефіцієнта стиснення. Таким чином, при стисненні вимірювальної інформації спостерігається значне збільшення коефіцієнта стиснення зі збільшенням обсягу первинного блока, що пояснюється структурованістю цього виду інформації, де деякі рядки відрізняються тільки кількома символами, за рахунок чого досягається дуже

великий коефіцієнт стиснення. Цей вид стиснення можна також рекомендувати і для передачі вимірювальної інформації по каналах зв'язку, але з блоками, що не перевищують 1 024 байт, тому що при подальшому збільшенні обсягу блока час стиснення значно зростає.

Як відомо, метод стиснення LZW (рис. 3, крива 2) є словниковим префіксним методом. Тому, чим буде більше слів, які повторюються, у вхідному блоці, що стискається, тим більше зростає коефіцієнт стиснення. Особливо значне збільшення K_c відбувається, починаючи з блока обсягом 128 байт. Стиснення методом LZW можна рекомендувати для застосування в АСК ТП та системах передачі даних.

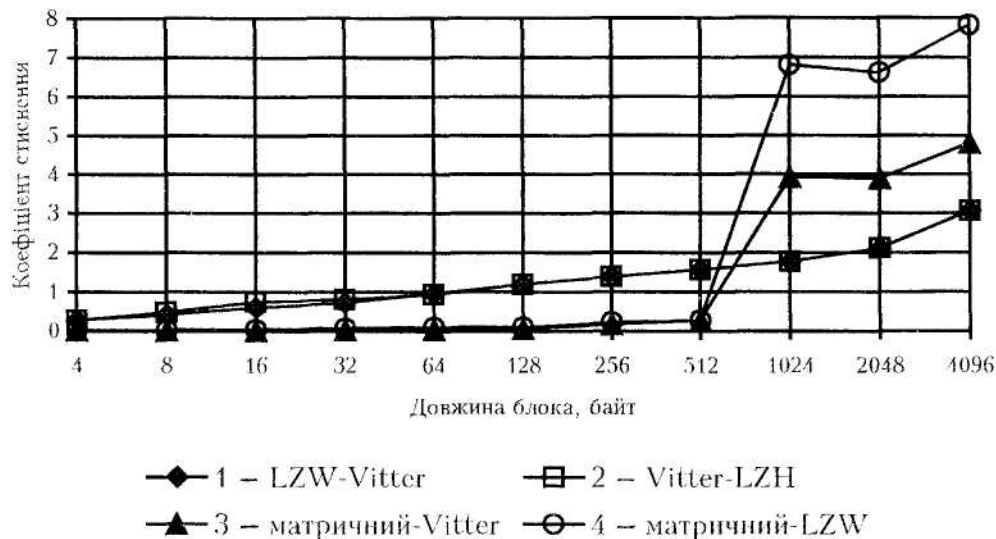


Рис. 4

При стисненні вимірювальної інформації методом Vitter (рис. 3, крива 3), збільшення коефіцієнта стиснення спостерігається тільки починаючи з розміру блока 1 024 байти. Але коефіцієнт K_c при цьому буде менший, ніж при інших методах стиснення. Можна застосовувати даний метод стиснення при надходженні вимірювальної інформації від великої кількості датчиків за малі проміжки часу.

Результати стиснення матричним методом вимірювальних даних показані на рис. 3 кривою 4. Інформація цього типу добре піддається матричному стисненню вже з розміру блока 512 байт. Це пов'язано з пошуком оптимального розміру матриці даних. Як видно з рисунка, подальша адаптація до вхідних даних дозволяє значно збільшити коефіцієнт стиснення. Цей метод можна рекомендувати для застосування в АСК ТП та при передачі структурованих даних по каналах зв'язку з мінімальним блоком не менше ніж 512 байт.

При каскадних методах здійснюється послідовне стиснення двома методами. При цьому на різних каскадах для стиснення використовуються різні алгоритми. Дослідження каскадних методів здійснювалося на потоках даних із використанням комбінацій чотирьох видів розглянутих класичних алгоритмів стиснення.

Залежність коефіцієнта стиснення від первинного потоку вимірювальної інформації при використанні каскадного LZW-Vitter методу показана на рис. 4 кривою 1, а Vitter-LZH методу – кривою 2. Як випливає з цих кривих, коефіцієнт стиснення для вказаних методів збігається майже для всіх обсягів вхідного потоку даних і є не дуже великим, але монотонно збільшується зі збільшенням обсягу вхідного потоку, що стискається.

Каскадний матричний-Vitter метод (рис. 4, крива 3) дає непогане значення коефіцієнта стиснення для вимірювальних даних, у порівнянні з розглянутими вище двома каскадними методами.

Крива 4 на рис. 4 відображає залежність коефіцієнта стиснення для каскадного матричний-LZW методу. Він має найбільший коефіцієнт стиснення, у порівнянні з усіма каскадними методами.

Оцінка швидкодії різних алгоритмів стиснення

Метою даного дослідження є оцінка швидкодії розглянутих вище алгоритмів стиснення для вимірювальної інформації та різних параметрів первинних інформаційних потоків, що стискаються.

Тестування здійснювалося на процесорах AMD586-133, Intel Pentium-200. Для виключення впливу розміру і швидкості кеша другого рівня на швидкість роботи алгоритму кеш-пам'ять системи відключалася. Для тестування використовувалися фрагменти даних довжиною 4 096 байт, при цьому вимір робився в циклі 1 000 разів для підвищення точності результатів.

На рис. 5 показана залежність часу стиснення від довжини блока вимірювальної інформації, що стискається, для різних класичних методів стиснення: крива 1 – для методу LZH, 2 – LZW, 3 – Vitter, 4 – матричного.

Як випливає з рис. 5, час стиснення вимірювальної інформації методом LZH (крива 1) і методом LZW (крива 2) майже збігається. При чому, чим більша довжина блока, тим більше часу витрачається на його стиснення. Особливо це помітно, починаючи з блока довжиною 128 байт.

Залежність часу стиснення від розміру блока вимірювальної інформації для методу Vitter показана на рис. 5 кривою 3. З розглянутих алгоритмів, даний метод стиснення дає найменший час стиснення. Незначний час стиснення пояснюється тим, що процедура стиснення, яка забезпечує побудову дерева, виконується за постійний час не залежно від розміру блока.

Залежність часу стиснення від довжини блока вимірювальної інформації для матричного методу показана на рис. 5 кривою 4. Як випливає з рисунка, матричний метод виконує стиснення різних за довжиною інформаційних блоків практично за однаковий час.

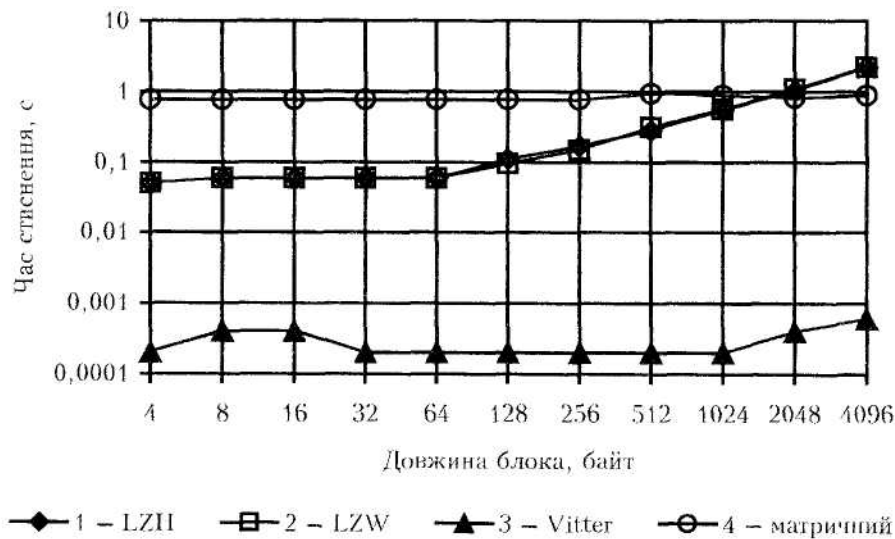


Рис. 5

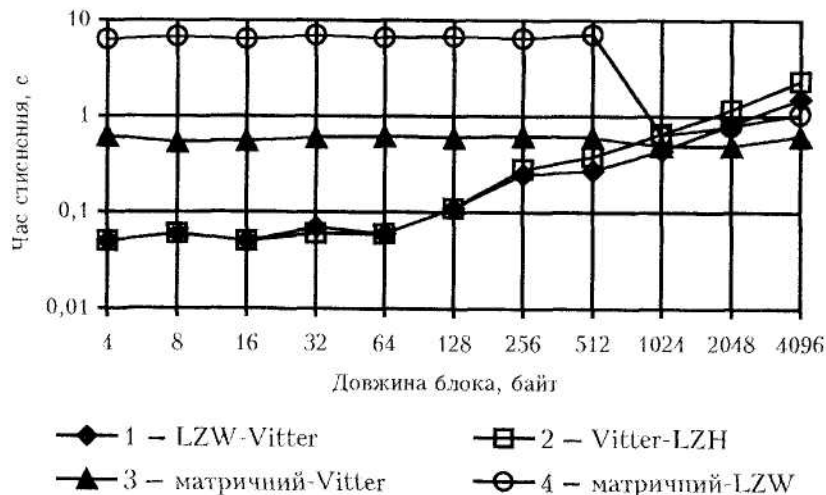


Рис. 6

Залежність часу стиснення від довжини блока вимірювальної інформації, що стискається, для каскадних методів показана на рис. 6, де крива 1 відповідає методу LZW-Vitter; 2 – Vitter-LZH; 3 – матричний-Vitter; 4 – матричний-LZW. Як випливає з рис. 6, час стиснення для перших двох методів, LZW-Vitter (крива 1) та Vitter-LZH (крива 2), майже збігається для інформаційних блоків будь-якої довжини і помітно збільшується для блоків, починаючи з 128 байт. У той же час, для каскадного матричний-Vitter методу (крива 3) час стиснення майже не залежить від довжини блока, але для блоків довжиною до 1 024 байт значно перевищує час стиснення першими двома методами. За часом стиснення каскадний матричний-LZW метод (крива 4) значно перевищує всі інші методи, особливо при стисненні інформаційних блоків довжиною до 512 байт.

ВИСНОВКИ

В роботі запропоновано методику проведення досліджень ефективності методів стиснення вимірювальної інформації, яка широко використовується в АСК ТП. Розроблено алгоритми, що реалізують стиснення даних деякими методами, і алгоритм основної вимірювальної процедури, за допомогою якого одержують основні показники ефективності алгоритмів стиснення та систематизують отримані результати. За цими алгоритмами здійснена програмна реалізація на мові Object Pascal у системі програмування Borland Delphi 4.0 Client/Server для операційних систем Windows 98 і Windows NT.

Виконано дослідження залежності коефіцієнта стиснення від довжини блока для різних методів. Проведено порівняльну оцінку методів за значеннями коефіцієнта стиснення на різних інформаційних потоках.

Виходячи з результатів проведених досліджень, можна дати наступні рекомендації по реалізації стиснення вимірювальної інформації в АСК ТП та у системах збору, передачі й обробки інформації.

Так, з точки зору ступеня стиснення, для інформаційного потоку вимірювальних даних, найкраще підходять LZW, LZH методи. Також непогані результати дає використання матричного методу, але коефіцієнт стиснення при цьому дещо менший, ніж при використанні методів LZW і LZH. З каскадних методів стиснення надати перевагу можна тільки методу, де на першому каскаді використовується матричний метод, а на другому – метод Vitter.

У той же час, з точки зору часу стиснення інформаційного потоку вимірювальних даних, найкращі показники мають: з класичних методів стиснення – Vitter, LZH і LZW. Особливо метод Vitter, який має дуже незначний час стиснення інформаційних потоків, що практично не залежить від довжини інформаційного блока. З каскадних методів стиснення надати перевагу можна LZW-Vitter і Vitter-LZH, швидкодія яких практично дорівнює швидкодії методів LZW і LZH.

Таким чином, до найбільш ефективних методів стиснення вимірювальної інформації, з точки зору ступеня стиснення та швидкодії, можна віднести такі класичні методи стиснення, як: LZW і LZH, каскадні методи LZW-Vitter і Vitter-LZH та каскадні методи матричний-Vitter і матричний-LZW для інформаційних масивів довжиною від 1 024 байт.

ЛІТЕРАТУРА:

1. Кричевский Р.Е. Сжатие и поиск информации. – М.: Радио и связь, 1989. – 168 с.
2. Курицын С.А. Методы адаптивной обработки сигналов передачи данных. – М.: Радио и связь, 1988. – 144 с.
3. Чернега В.С. Сжатие информации в компьютерных сетях. – Севастополь: Сев.ГТУ, 1997. – 214 с.
4. Ziv J., Hershkovitz Y. Another Look at Universal Data Compression // Proc. IEEE Int. Sympos.on Information Theory. – Trondheim, Norway, 1994. – P. 11.

ЖУРАКОВСЬКИЙ Ярослав Юрійович – аспірант Національного технічного університету України “Київський політехнічний інститут”.

Наукові інтереси:

- автоматизовані системи керування технологічними процесами;
- обробка інформації.

ЗАПОЛЬСЬКИЙ Сергій Михайлович – студент 6-го курсу Національного технічного університету України “Київський політехнічний інститут”.

Наукові інтереси:

- програмування;
- обробка інформації.