

## АЛГОРИТМ НАДІЙНОГО ЗБЕРІГАННЯ ДАНИХ НА ОСНОВІ МОДУЛЯРНИХ КОРИГУЮЧИХ КОДІВ

Яцків Н.Г.<sup>1)</sup>, Шаршин В.С.<sup>2)</sup>, Гевко Я.І.<sup>3)</sup>

Тернопільський національний економічний університет

<sup>1)</sup> к.т.н., доцент; <sup>2,3)</sup> магістрант

### I. Постановка задачі

На даний час кількість підключених пристроїв Інтернет речей (IoT) вже перевершила загальну кількість населення на Землі. Інтеграція IoT та парадигми хмарних обчислень розширює сферу діяльності та можливості надання послуг. Хмарні сервіси не лише отримують доступ до базових вузлів IoT та надають їм хмарні послуги, але також можуть створювати додаткові розширені сервіси на основі даних, зібраних з вузлів IoT.

Проте, мережі IoT знаходяться далеко від хмари, яка зазвичай доступна через Інтернет. Деякі служби IoT вимагають швидкого реагування та великої кількості попередньої обробки та фільтрації даних, а також можуть мати підвищені вимоги до безпеки та конфіденційності. Для таких випадків між хмарою та IoT-пристроями використовують проміжне програмне забезпечення, також відоме як «туман».

IoT пристрої генерують значні обсяги даних, наприклад, датчики в одному автономному автомобілі можуть створювати 4000 гігабіт (Gb) даних на день; літак Airbus A380 Engine створює 1 петабайт (Pb) даних про рейс з Лондона до Сінгапуру; датчики безпеки в процесі гірничодобувної діяльності можуть створювати до 2.4 терабіт (Tb) даних щохвилини; датчики в одному розумному будинку можуть виробляти до 1 гігабайту (Gb) інформації на тиждень [1].

Враховуючи, що мережі IoT генерують великі обсяги даних, які потребують попередньої обробки та зберігання розробка мережного сховища даних є актуальною задачею.

### II. Мета роботи

Метою роботи є розробка алгоритму надійного зберігання даних на основі модулярних коригуючих кодів.

### III. Алгоритм надійного зберігання даних

Пропонований алгоритм надійного зберігання даних базується на модулярних коригуючих кодах, принцип роботи яких полягає в наступному. Пакет даних в двійковому коді, який підлягає зберігання, розбивається на  $k$  блоків однакової довжини (один або два байти) [2]:

$$a_j^1 \dots a_3^1 a_2^1 a_1^1 a_0^1, a_j^2 \dots a_3^2 a_2^2 a_1^2 a_0^2, a_j^i \dots a_3^i a_2^i a_1^i a_0^i, \dots, a_j^k \dots a_3^k a_2^k a_1^k a_0^k,$$

де  $a^i$  – розряд даних в двійковому коді,  $j = 4, 8$ .

Значення перевірного символу обчислюється за формулою:

$$x_{k+1} = \left| v_1 \cdot x_1 + v_2 \cdot x_2 + \dots + v_i \cdot x_i + \dots + v_k \cdot x_k \right|_p, \quad (1)$$

де  $v_i$  – коефіцієнти, взаємно прості з модулем  $p$ ;  $x_i$  – дані в двійковій або десятковій системі числення.

При цьому блок даних розділяється на частини  $x_i$ , кількість частин дорівнює кількості дисків, тобто кожна частина даних і перевірочний символ зберігається на окремому диску.

Модулярні коригуючі коди забезпечують виправлення помилок в одному блоці даних, використовуючи один перевірочний символ [2].

### Висновок

У роботі розроблено алгоритм надійного зберігання даних на основі модулярних коригуючих кодів, який забезпечує відновлення даних при виході з ладу одного з дисків.

### Список використаних джерел

1. What is Big Data? [Електронний ресурс]. – Режим доступу: <http://www.royinfo.xyz/what-is-big-data/>
2. Яцків В.В. Виявлення та виправлення багатократних помилок на основі модулярних коректуючих кодів / В.В. Яцків // Інформаційні технології та комп'ютерна інженерія. – 2015. – Том 33, №2. – С.77-82.