

## АНАЛІЗАТОР СКІНЧЕННИХ ЦИФРОВИХ АВТОМАТІВ

*Для рішення задачі синтезу детермінованого скінченного автомата запропоноване використання діаграм двійкових рішень. Це дозволяє істотно поліпшити масштабованість методу й представити структуру автомата у вигляді бінарного дерева – структури даних, зручної для наступного етапу аналізу. Використання на етапі аналізу модифікованого мультиагентного методу мурашиних колоній (МК) та імовірнісного підходу дозволило визначити оптимальні шляхи обходження графу й імовірності перебування автомата в кожному зі станів.*

**Постановка задачі.** Предметом дослідження даної роботи є розробка ефективного аналізатора скінчених цифрових автоматів.

Для досягнення поставленої мети були вирішені такі завдання: досліджені методи аналізу цифрових автоматів; проведений аналіз методів синтезу цифрових автоматів; визначені вимоги до аналізаторів цифрових автоматів; програмно реалізовані обрані й розроблені методи аналізу й синтезу.

Проблема аналізу полягає у визначенні подій, що представляються в автоматі множиною вихідних сигналів (автомати Мілі) або множиною внутрішніх станів (автомати Мура). Виходячи з того, що кожен автомат Мура можна інтерпретувати як автомат Мілі, досить навчитися аналізувати лише автомати Мілі.

Нині найчастіше використовується метод аналізу скінчених автоматів [1], що повертає лише автоматні слова (регулярні вирази у відображенні автомата). При цьому не передбачається можливість виявлення альтернатив знайденим словам.

Для досягнення кращих результатів аналізу скінченного автомата, підвищення швидкодії методу й одержання додаткової інформації пропонується використовувати метод МК.

**Модифікація методу МК для розв'язання задачі аналізу детермінованого скінченного автомата.** Метод МК заснований на моделюванні взаємодії декількох штучних аналогів мурах, що представляються програмно у вигляді інтелектуальних агентів, що є членами великої колонії. Базова ідея методу [2] полягає в розв'язанні оптимізаційної задачі шляхом застосування непрямого зв'язку між автономними агентами, що моделюються. Такі агенти, переміщаючись по графу рішень, допомагають іншим агентам у подальшій оптимізації рішення.

Дослідження особливостей методу [3] дали підстави для того, що деякі фундаментальні положення методу МК можуть служити базисом у розв'язанні задачі пошуку оптимальних маршрутів усередині графа автомата. Інформація, яка отримана після завершення даного пошуку, дає можливість визначити найбільш короткі шляхи обходу графа, необхідні для досягнення істинного або хибного результату. Це, у свою чергу, може привести до висновків щодо необхідності оптимізації початкової логічної функції.

Відповідно до вимог методу МК задача, що розв'язується, формально повинна представлятися у вигляді графа з набором вузлів і зв'язків між ними. Використання графа досить зручне, оскільки в результаті синтезу створюється структура даних типу дерево, яку легко можна представити у вигляді графа.

Адаптуючи логіку проблеми під систему понять методу МК, одержуємо такі відповідності:

- агенти, що моделюються, – символи алфавіту автомата (команди переходів);
- граф (дерево) – подання відображення автомата;
- ребра графа (грані) – переходи між станами автомата;
- вузли графа (вершини) – статичні стани автомата;
- феромон – частота виникнення ситуації переходу автомата;
- ітерація методу – цикл роботи автомата.

Спочатку необхідно проіндексувати всі вузли графа автомата. Тоді кожному вузлу буде відповідати число  $n$  в інтервалі  $[1, N]$ , де  $N$  – кількість вузлів графа автомата. У такому випадку вихідними даними буде виступати таблиця даних, кожна комірка якої містить інформацію щодо накопиченого феромону, а індекси комірки будуть відповідати номерам вихідного й поточного вузлів графа автомата. Відмінність від класичного формулювання методу полягає в тому, що для пошуку найкоротшого шляху вибирається тільки один початковий вузол – вершина дерева.

Метод містить такі кроки:

Крок 1. Ініціалізуємо матрицю феромонів  $T = \{\phi_{ij}^1, \phi_{ij}^2, \dots, \phi_{ij}^N\}$ , де  $N$  – кількість вузлів автомата;  $\tau_{ij}$  – ребра графа. У матриці номери рядків  $i$  будуть відповідати індексам попередніх вузлів, з яких був зроблений перехід у стан, що відповідає номерам стовпчиків  $j$ .

Крок 2. Виконується перебір елементів матриці феромонів. Для кожного циклу виконуються кроки 2.1, 2.2.

Крок 2.1. Перебираємо елементи структури  $V_m = \{s, q, p\}$ . Якщо  $q = i$  й  $p = j$ , де  $i$  та  $j$  – номери рядка й стовпчика матриці  $T$  відповідно, то збільшуємо кількість феромону на ребрі  $\tau_{ij}$  на задану величину феромону  $D_f$ .

Крок 2.2. Робимо випар феромона на всіх ребрах графа, використовуючи коефіцієнт випару  $\rho$ , за формулою:

$$\phi_j = \phi_j \cdot \rho. \quad (1)$$

Коли всі цикли розглянуті, перейти до кроку 3.

Крок 3. Ініціалізуємо деяке граничне значення  $t$ .

Крок 4. Виконуємо перевірку всіх елементів матриці феромонів  $T = \{\phi_{ij}^1, \phi_{ij}^2, \dots, \phi_{ij}^N\}$  щодо порогу  $\phi_{ij} > t$ .

Крок 5. Серед тих елементів матриці  $T$ , значення яких перевищили поріг, проводимо пошук послідовностей, що приводять із початкового стану графа в одне зі скінченних.

Переваги розробленого методу:

– така властивість інтелектуальних систем, як уміння навчатися, дозволяє при надходженні нових вихідних даних у систему використовувати досвід накопичених за попередній час роботи знань, що полегшує повторні обчислення, які в цьому випадку повинні проводитися тільки над новими даними;

– метод дозволяє раціонально визначати припустимі рішення, змінюючи значення порога  $t$  і щораз звужуючи або розширюючи область результатів, що одержуються залежно від задачі, що розв'язується.

Недолік методу – необхідність проводити додатковий пошук повних послідовностей переходів.

Швидкість роботи методу МК лінійно залежить від кількості виконаних переходів, оскільки при обчисленнях на кожному перехід затрачається одна операція додавання (накопичення феромону на ребрі графа). До того ж спостерігається квадратична залежність швидкості методу від вибраного як ітерація проміжку часу (у даному прикладі – кількості циклів роботи). Відомо, що закінчення кожної ітерації супроводжується відновленням феромону на ребрах графа, а це, при кількості вузлів графа  $n$ , буде становити  $n^2$  операцій множення феромону на коефіцієнт випару.

Проаналізувавши переваги й недоліки методу МК у застосуванні до завдання пошуку оптимальних маршрутів, зроблені висновки про доцільність його використання для знаходження оптимальних маршрутів. Модифікація методу, що дає можливість випаровувати феромон пропорційно довжині пройденого агентом маршруту, дозволяє оптимізувати знаходження найкоротших маршрутів за критерієм кількості переходів. Дані про кількість переходів між станами автомата також є вхідними до використання методу визначення імовірності переходу автомата в той або інший стан.

**Застосування імовірнісного підходу до кількісного визначення інформації для обчислення імовірності переходу автомата в певний стан.** Найбільш відомим і широко застосовуваним на практиці є імовірнісний підхід до виміру інформації. На основі цього підходу розроблений великий розділ кількісної теорії інформації, що відповідно до імені його основоположника називається "теорією інформації Шеннона". Головною відмінністю імовірнісного підходу від комбінаторного є той факт, що він заснований на імовірнісних припущеннях щодо перебування якої-небудь системи в різних станах. При цьому загальна кількість елементів (мікростанів, подій) системи не враховується. За кількість інформації тут береться знята невизначеність вибору з множини можливостей, що мають у загальному випадку різну ймовірність [4].

Основна роль в імовірнісному підході належить ентропії множини імовірностей. Пропонуючи для виміру кількості інформації ентропійну міру, Шеннон керувався наступними міркуваннями. Припустимо, що є деяка множина можливих подій, імовірності здійснення яких  $p_1, p_2, \dots, p_n$ . Ці ймовірності відомі, але це – усе, що нам відомо щодо того, яка подія відбудеться. Необхідно знайти міру  $H$  того, наскільки велика ймовірність переходу системи до певної події. Для такої міри  $H$  висувається вимога: вона повинна мати такі три властивості:

- $H$  повинна бути безперервною відносно  $p_i$ ;
- якщо всі  $p_i$  рівні, то  $H$  повинна бути монотонно зростаючою функцією від  $n$ ;
- якщо вибір розпадається на два послідовних вибори, то первісна  $H$  повинна бути зваженою сумою індивідуальних значень  $H$  кожного з виборів.

У нашому випадку розрахунок ентропії  $H$  буде необхідний згодом для нормалізації матриці, що є вхідним параметром методу.

На виході методу МК можна одержати заключну матрицю феромонів, що наочно показує кількісну характеристику переходів між станами автомата. Після нормалізації матриці її комірки будуть представляти значення імовірностей перемикавання автомата між станами.

Правило нормалізації матриці таке: сума імовірностей переходів з будь-якого вузла дерева до дочірних вузлів повинна бути меншою або дорівнювати 1. Ця умова задовольняє формулу ентропії Шеннона. Після нормалізації матриці необхідно обчислити імовірності переходу автомата між станами.

Імовірність для кожного вузла дерева обчислюється як сума добутків імовірностей усіх ребер графа на шляху від початкової точки до необхідного вузла.

На підставі результатів дослідження розроблене програмне забезпечення, призначене для аналізу й синтезу скінченних цифрових автоматів. Функціональна схема програмної системи наведена на рис. 1.

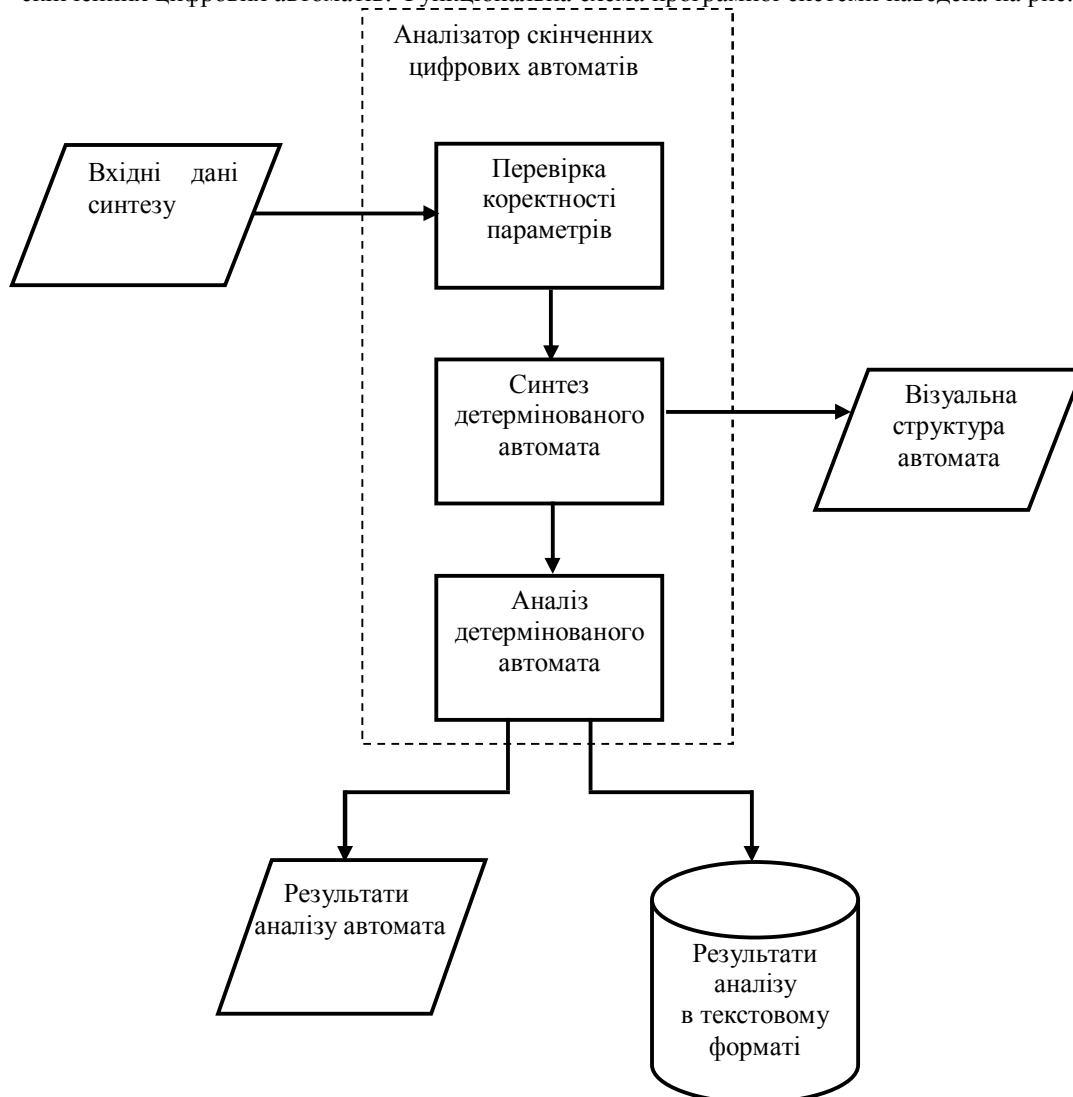


Рис. 1. Функціональна схема програмної системи

Система складається з таких модулів:

- лексичний аналізатор – призначений для розбору булевого виразу, що описує поведінку скінченного автомата. Результатом роботи модуля є подання булевої функції у вигляді бінарного дерева;
- синтезатор скінченного автомата – призначений для синтезу структури скінченного автомата у вигляді бінарного дерева. Вихідними даними для роботи модуля є булева функція, що представлена у вигляді дерева;
- модуль візуалізації структури скінченного автомата – призначений для візуального відображення структури синтезованого скінченного автомата;
- аналізатор скінченного автомата – виконує аналіз на основі методів МК та імовірнісного підходу;
- модуль відображення результатів аналізу – виконує функції інтерфейсу користувача й служить для подання результатів аналізу скінченного автомата в зручному для користувача вигляді.

Вхідними даними для програми є логічна функція, представлена в диз'юнктивній нормальній формі, що описує поведінку скінченного автомата. Потім здійснюється його синтаксичний розбір і у випадку успішного його виконання виконується синтез скінченного автомата. Синтезований цифровий автомат візуально надається користувачеві у вигляді графа, й починається операція його аналізу.

Вихідними даними програми є інформація про структуру побудованого детермінованого скінченного автомата, таблиця результатів аналізу за допомогою методу МК і список результатів аналізу за

допомогою імовірнісного підходу. Результати аналізу виводяться в текстовому й графічному вигляді на форму, представлену на рис. 2.

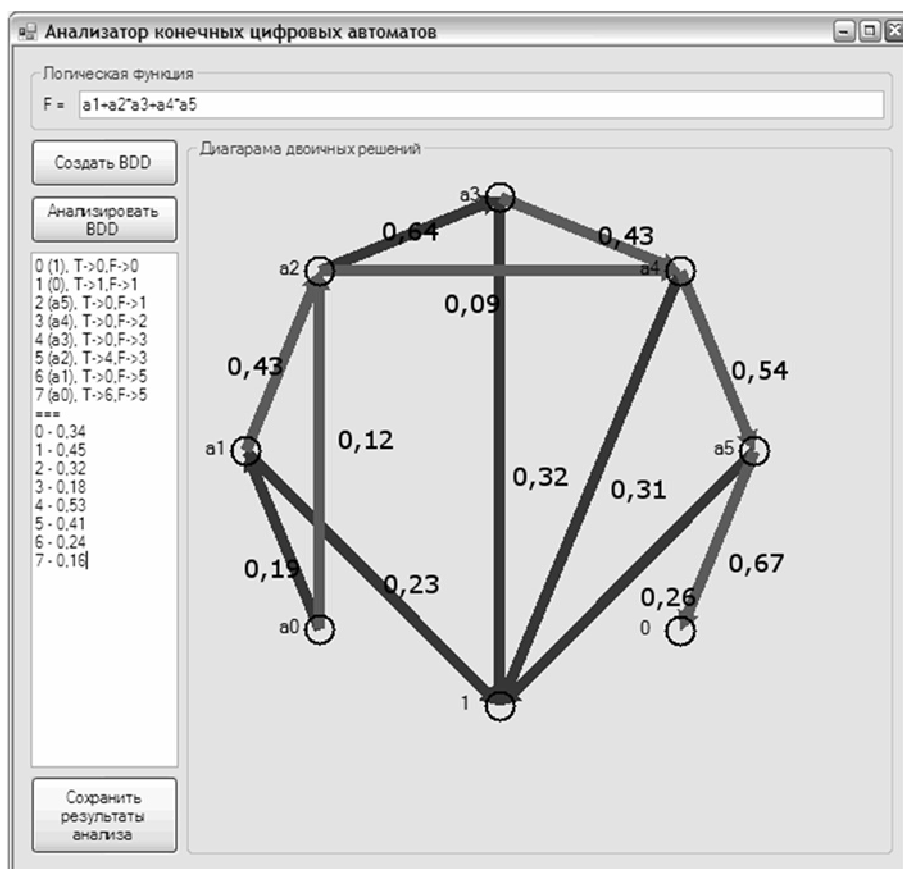


Рис. 2. Форма з результатами аналізу

**Висновки.** Наукова новизна роботи полягає в тому, що подальший розвиток одержали методи МК та імовірнісного аналізу, що дозволило: застосувати зазначені методи, а також діаграми двійкових рішень у задачі аналізу й синтезу цифрових автоматів; поліпшити масштабованість методу синтезу й знизити витрати обчислювальних ресурсів у порівнянні з існуючими методами аналізу.

Практична цінність роботи полягає в тому, що була створена програмна система синтезу моделі скінченного цифрового автомата за деякою заданою логічною функцією й наступним аналізом отриманого цифрового автомата. Для зображення моделі автомата як структура даних використовується бінарне дерево. Система реалізована мовою C# пакета Microsoft Visual Studio 2008. Для забезпечення її нормальної роботи необхідна апаратна платформа з мінімальним обсягом оперативної пам'яті 512 Мб, не менш 10 Мб вільного дискового простору, а також наявність встановленого на комп'ютері пакета .NET 3.5. До операційної системи вимог не пред'являється.

#### ЛІТЕРАТУРА:

1. Han J. Data mining: concepts and techniques / J.Han, M.Kamber. – San Francisco : Morgan-Kaufmann Academic Press, 2001. – 387 p.
2. Agrawal R. Mining sequential patterns / R.Agrawal, R.Srikant // Proceedings of the 11<sup>th</sup> International Conference on Data Engineering. – Taiwan, 2001. – P. 3–14.
3. Штучні імунні системи і їхнє застосування / за ред. Д.Дасгупты ; пер. з англ. А.А. Романюхи. – М. : ФИЗМАТЛИТ, 2006. – 344 с.
4. Wang W. A complex artificial immune system and its immunity / W.Wang, S.Gao, F.Li, Z.Tang // International Journal of Computer Science and Network Security. – 2008. – № 12. – P. 161–169.

СЕРДЮК Сергій Микитович – кандидат технічних наук, доцент кафедри програмних засобів Запорізького національного технічного університету.

Наукові інтереси:

– методи отримання аналітичної інформації;  
– комп'ютерно-інформаційні технології.  
E-mail: serdjuk@zntu.edu.ua

Подано 21.01.2010

Сердюк С.М. Аналізатор скінченних цифрових автоматів  
Сердюк С.Н. Аналізатор конечных цифровых автоматов

УДК 519.713

**Аналізатор конечных цифровых автоматов / С.Н. Сердюк**

Для решения задачи синтеза детерминированного конечного автомата предложено использование диаграмм двоичных решений. Это позволяет существенно улучшить масштабируемость метода и представить структуру автомата в виде бинарного дерева – структуры данных удобной для последующего этапа анализа. Использование на этапе анализа модифицированного мультиагентного метода муравьиных колоний (МК) и вероятностного подхода позволило определить оптимальные пути обхождения графа и вероятности пребывания автомата в каждом из состояний.

УДК 519.713

The suggestion to solve the deterministic finite automaton synthesis problem is to use binary solution diagrams. Significant improvements achieved are extended scalability and further analysis capabilities while representing the automation data structure as binary tree nodes. The analysis works involves the modified Multi Agents Ant Colony method and probability approaches to optimize graph-states paths as well as automation state probability data acquisition.