

УДК 519.713.1

**к.т.н Потапова К.Р., студент Богаченко А.К.**

**Національний технічний університет України  
«Київський політехнічний інститут»**

## **ПРОЕКТУВАННЯ ТА ОПТИМІЗАЦІЯ АЛГОРИТМУ КЕРУЮЧОЇ ЧАСТИНИ РЕАКТИВНОЇ СИСТЕМИ**

### **Abstract**

*K.R. Potapova, assoc. prof., PhD; A.K. Bogachenko, student*

*The design of the control unit of a reactive algorithm, in the framework of automata approach to the specification, and design of open systems is considered. The main problem is to synthesize an automaton, providing the system operation in accordance with its specification for any admissible behavior of the environment. The basic notion in the formulation of this problem is the correctness of the composition of interacting automata.*

### **Вступ**

Під реактивними системами розуміються системи, що постійно взаємодіють зі своїм оточенням. Прикладами таких систем можуть бути системи управління технологічними процесами, системи управління літальними апаратами, системи управління дорожнім рухом та інші. При проектуванні таких систем необхідно гарантувати точну відповідність алгоритму управління всім вимогам до його функціонування. В основі підходу, що розглядається до такого проектування алгоритму лежить специфікація функціональних вимог до нього у мові логіки першого порядку і формальний перехід від специфікації до процедурного поданням алгоритму, що називається синтезом.

### **Постановка задачі**

Специфікація вимог до функціонування реактивного алгоритму складається з двох частин: специфікації керуючої частини алгоритму і специфікації середовища, з якою він взаємодіє. Як математична модель, що визначає семантику кожної з цих частин, використовується скінченний автомат. Будемо називати ці автомати відповідно керуючим та операційним. Основна проблема полягає в тому, щоб синтезувати керуючий автомат, що

забезпечує функціонування системи відповідно до її специфікації, при будь-якій допустимій поведінці зовнішнього середовища. Традиційно ця проблема формулюється як проектування «відкритої системи», що відображає неможливість в процесі проектування впливати на специфікацію середовища. Вперше практичні методи синтезу відкритої системи були розглянуті в роботах [1] та [2]. У них задача синтезу формулюється як знаходження вигрешної стратегії в нескінченній грі між середовищем та системою. Така стратегія представляється у вигляді нескінченного розміченого дерева. Для опису всіх стратегій, які задовольняють специфікацію, будується автомат Рабина над нескінченними деревами [3], для якого множина всіх розпізнаваних ним нескінченних дерев відповідає всім стратегіям, які реалізують вихідну специфікацію. Перевірка реалізованості специфікації зводиться до перевірки непорожності отриманого автомата Рабина. У цілому така процедура синтезу вимагає часу, що виражається у вигляді подвійної експоненти від розміру специфікації. В [4] поліпшено оцінки складності за рахунок спрощення мови специфікації, в якості якої розглядається підмножина GR (1) темпоральної логіки з лінійним часом. У роботах [5, 6] запропонований інший підхід, заснований на понятті узгодженості керуючого автомата з операційним (середовищем). Задача перевірки узгодженості автоматів розв'язується на рівні їх специфікацій логічною мовою. При цьому складові алгоритму специфікуються і синтезуються як неініціальні системи. У цій роботі пропонується підхід до проектування ініціальної системи, що не вимагає проведення узгодження неініціальних специфікацій.

### Основна частина

Розглядаються часткові недетерміновані  $X$ – $Y$ -автомати Мура. Такий автомат  $A$  визначається п'ятіркою  $\langle X, Y, Q, \chi_A, \mu_A \rangle$ , де  $X, Y, Q$  — скінченні множини відповідно вхідних символів, вихідних символів і станів, а  $\chi_A : Q \times X \rightarrow 2^Q$  і  $\mu_A : Q \rightarrow Y$  — усюди визначені функції переходів і виходів. Якщо  $|X| = 1$ , то автомат  $A$  називається *автономним*  $Y$ -автоматом, який визначається четвіркою  $\langle Y, Q, \chi_A, \mu_A \rangle$ . Функція переходів автомата має вигляд  $\chi_A : Q \rightarrow 2^Q$

Оскільки автоматні моделі визначають поведінку алгоритму, що працює потенційно нескінченно, розглядаються автомати над нескінченними вхідними послідовностями, що відповідає поняттю циклічного автомата.

$X$ – $Y$ -автомат  $A = \langle X, Y, Q, \chi, \mu \rangle$  називається *циклічним*, якщо для всіх  $q \in Q$  існують такі  $q_1, q_2 \in Q$  та  $x_1, x_2 \in X$ , що  $q_1 \in \chi(q, x_1)$  і  $q \in \chi(q_2, x_2)$ .

Взаємодіючі автомати утворюють структуру, в якій входи і виходи одного з них з'єднані відповідно з виходами і входами іншого. Характер цієї взаємодії визначається наступним поняттям композиції автоматів.

Права циклічна композиція автоматів  $A$  і  $B$  являє собою максимальний циклічний підавтомат автономного  $Z$ -автомата Мура  $C = \langle Z, Q_C, \chi_C, \mu_C \rangle$ , де  $Z = Y \times X$ ,  $Q_C = Q_B \times Q_A$ , а функція переходів  $\chi_C$  і функція виходів  $\mu_C$  визначаються наступним чином. Для всіх  $a \in Q_A$ ,  $b \in Q_B$   $\chi_C(a, b) = \{(a', b') \mid b' \in \chi_B(b, \mu_A(a)) \text{ а } a' \in \chi_A(a, \mu_B(b'))\}$ ,  $\mu_C(a, b) = (\mu_A(a), \mu_B(b))$ .

У випадку частковості взаємодіючих автоматів виникає проблема забезпечення коректності їх спільного функціонування, тобто виключення такої ситуації, при якій в процесі функціонування автоматів перехід в одному з них під впливом вхідного сигналу, що надходить з іншого, буде не визначений. Проектування керуючого автомата здійснюється наступним чином. Синтезують композицію автоматів, яка розглядається як керуючий автомат з додатковою частковістю, отриманою за рахунок його взаємодії з середовищем. Ця частковість використовується для оптимізації автомата шляхом відповідного довизначення. Однак довільне довизначення автомата можна здійснювати тільки в тому випадку, якщо відповідна композиція задовольняє наступну умову коректності.

Нехай  $b_1, \dots, b_m$  — відмітки всіх тих станів, у які автомат  $B$  переходить із стану  $b$  під дією символу  $y_0 = \mu_A(a)$ . Композиція автоматів  $A$  і  $B$  **коректна в стані  $(a, b)$**  тоді і тільки тоді, коли **не існує**  $b_i \in \{b_1, \dots, b_m\}$ , такого, що або перехід в  $A$  із стану  $a$  під дією  $\mu_B(b_i)$  не визначений ( $\chi_A(a, \mu_B(b_i)) = \emptyset$ ), або для всіх  $a_i \in \chi_A(a, \mu_B(b_i))$  стани  $(a_i, b_i)$  не належать композиції.

Для перевірки коректності композиції зручніше користуватись поняттям паралельної циклічної композиції.

**Паралельна** циклічна композиція автоматів  $A$  і  $B$  — це максимальний циклічний підавтомат прямого добутку  $\Sigma$ -автоматів  $A$  і  $B$ . Відомо що права циклічна композиція автоматів  $A$  і  $B$  співпадає з паралельною циклічною композицією автоматів  $A$  і  $B^*$ , де  $B^*$  — лівий зсув за входом автомата  $B$ .

*Лівим зсувом*  $X \times Y$ -надслова  $l = (x_1, y_1)(x_2, y_2)(x_3, y_3) \dots$  називається надслово  $l^* = (x_2, y_1)(x_3, y_2) \dots$ . Нехай  $B$  — циклічний  $X - Y$ -автомат, а  $W_B$  — множина всіх допустимих для нього  $X \times Y$ -надслів. Існує циклічний  $X - Y$ -автомат  $B^*$ , у котрого множина всіх допустимих  $X \times Y$ -надслів співпадає з  $W^*_B = \{l^* \mid l \in W_B\}$ . Всякий такий автомат будемо називати *лівим зсувом за входом автомата  $B$* .

Композиція автоматів  $A$  і  $B^*$  **коректна**, якщо для кожного її стану  $(a, b)$  проєкція  $\Sigma(b)$  на  $X$  співпадає з проєкцією  $\Sigma(a, b)$  на  $X$ .

Якщо в деякому стані композиції умова коректності не виконується, то відповідний стан керуючого автомата видаляється, що може призвести до збільшення частковості станів, що лишилися. Після цього слід перевірити коректність нової композиції. Процес перевірки коректності композиції і видалення станів закінчується, якщо після чергового видалення стану композиція стане коректною, або будуть видалені всі стани, що відповідає неможливості реалізації керуючого автомата.

## Висновок

У запропонованому підході до проектування відкритої системи зі специфікацією середовища узгоджується тільки ініціальна частина неініціальної специфікації керуючого компонента реактивного алгоритму, що часто призводить до істотного підвищення ефективності процедури проектування.

## Література

1. *Abadi M., Lamport L., Wolper P.* Realizable and unrealizable concurrent program specifications // LNCS. – 1989. – v. 372. – P. 1-17.
2. *Pnueli A., Rosner R.* On the synthesis of a reactive module // Proc. 16<sup>th</sup> Annual ACM Symp. On Principles of Programming Languages. – ACM, 1989. – P. 179-190.
3. *Thomas W.* Automata on infinite objects // Handbook of theoretical computer science, ed. J.van Leeuwen. –Amsterdam: Elsevier Sci. Publ., 1990. – P.134-191.
4. *Piterman N., Pnueli A., Sa`ar Y.* Sythesis of reactive(1) designs // Proc. Conf. on Verification, Model Checking and Abstract Interpretation. – 2006. – P. 364-380.
5. *Чеботаев А.Н.* Взаимодействие автоматов // Кибернетика и системный анализ. – 1991. –№6 – С. 17-29.
6. *Чеботаев А.Н.* Общий метод проверки согласованности взаимодействующих автоматов с конечной памятью // Кибернетика и системный анализ. – 1999 – №6. – С.25-37.