

*Баркалов О. О., д.т.н., професор, професор
кафедри інформаційних технологій
Бабаков Р. М., д.т.н., доцент, доцент
кафедри інформаційних технологій*

ТЕХНОЛОГІЇ ОБРОБКИ ДАНИХ В ПРОЕКТУВАННІ ЦИФРОВИХ СИСТЕМ КЕРУВАННЯ

Донецький національний університет імені Василя Стуса, м. Вінниця

Одним із різновидів аналізу даних є математична обробка великих масивів даних із використанням високорівневих мов програмування. В даній сфері значиме місце займає бібліотека NumPy, яка входить до складу мови програмування Python [1, 2]. Високошвидкісна реалізація функцій і методів обробки багатомірних масивів разом, простий синтаксис та кросплатформовість роблять NumPy високоефективним інструментом вирішення різного роду прикладних задач.

В даній роботі пропонується використовувати можливості бібліотеки NumPy для представлення і обробки даних, пов'язаних із процесом синтезу цифрових пристроїв керування, зокрема мікропрограмних автоматів (МПА) з операційним перетворенням кодів станів [3]. Основу синтезу схем таких автоматів становить пошук так званого формального рішення задачі алгебраїчного синтезу, в результаті якого кожен стан автомата отримує певний числовий код, а кожному мікропрограмному переходу ставиться у відповідність певна операція переходу (ОП), така, що забезпечує перетворення коду стану у код стану переходу [4].

Складність вирішення даної задачі полягає у тому, що кодування станів та вибір операцій переходів не є очевидними із структури алгоритму керування, що імплементується схемою МПА і припускає наявність множини формальних рішень. Пошук формальних рішень задачі алгебраїчного синтезу можливий завдяки частковому перебору варіантів кодування станів і переходів, в процесі якого необхідно щоразу перевіряти – чи є поточне кодування формальним рішенням задачі. Якщо в автоматі існує перехід зі стану a_i в стан a_j , причому стану a_i наданий код $K(a_i)$, стану a_j – код $K(a_j)$, а переходу між даними станами зіставлена деяка ОП, то необхідно перевірити, чи дійсно виконання даної ОП над кодом $K(a_i)$ дає в результаті значення $K(a_j)$. Подібні перевірки необхідні для кожного переходу автомата, тому аналіз отримання формального рішення задачі алгебраїчного синтезу зводиться до послідовного виконання множини арифметико-логічних операцій та перевірки правильності їх виконання. Щоразове виконання даної процедури потребує значних витрат часу і знижує ефективність методів пошуку формального рішення задачі алгебраїчного синтезу МПА.

Для спрощення процесу аналізу отримання формального рішення пропонується наступний підхід.

Представимо множину переходів МПА у вигляді квадратної матриці, в якій рядки і стовпці утворені усіма можливими кодами станів. Приклад матриці переходів для трирозрядних двійкових кодів станів, що приймають значення від 0 до 7, наведений на рис. 1. Одиниця на перетині рядка і стовпця означає, що існує перехід зі стану, записаного у відповідному рядку, в стан, записаний у відповідному стовпці. Порожня комірка означає відсутність переходу.

		a ₂	a ₅	a ₃	a ₆	a ₄	a ₀		a ₁
		0	1	2	3	4	5	6	7
a ₂	0			1					
a ₅	1				1				
a ₃	2					1			
a ₆	3						1		
a ₄	4						1		
a ₀	5								1
	6								
a ₁	7	1	1						

Рисунок 1 – Матриця переходів

Матриця переходів не містить інформації про вхідні та вихідні сигнали автомата. З матриці лише видно кодування станів та між якими станами здійснюються переходи. Наприклад, перехід зі стану a_1 відбувається в два різні стани, тобто є умовним переходом, інші переходи є безумовними. Також видно, що код 6 для кодування станів не використовується.

Нехай задані дві ОП: «+1» та «+2». Представимо дані ОП у вигляді матриць ОП, подібних матриці переходів (рис. 2).

	0	1	2	3	4	5	6	7
0		1						
1			1					
2				1				
3					1			
4						1		
5							1	
6								1
7	1							

	0	1	2	3	4	5	6	7
0			1					
1				1				
2					1			
3						1		
4							1	
5								1
6	1							
7		1						

Рисунок 2 – Матриці операцій переходів для операцій «+1» та «+2»

Одиниця на перетині рядка і стовпця означає, що за допомогою даної операції код, вказаний у відповідному рядку, перетворюється у код, вказаний у

відповідному стовпці. Функціональний характер подібних відношень унеможливорює появу двох одиниць в одному рядку матриці ОП.

Об'єднаємо матриці, зображені на рис. 2, в суміщену матрицю ОП, зображену на рис. 3.

	0	1	2	3	4	5	6	7
0		1	1					
1			1	1				
2				1	1			
3					1	1		
4						1	1	
5							1	1
6	1							1
7	1	1						

Рисунок 3 – Суміщена матриця операцій переходів

Розмір суміщеної матриці співпадає з розміром матриць ОП та матриці переходів. Вміст кожної комірки формується зі значень відповідних комірок матриць ОП (рис. 2) за наступним принципом: якщо хоча б в одній з матриць ОП відповідна комірка містить одиницю, комірка суміщеної матриці дорівнює одиниці; в іншому випадку комірка залишається порожньою.

Перевіримо, чи можна отримати формальне рішення задачі алгебраїчного синтезу при обраних кодах станів та операціях переходів. Для цього достатньо виконати зіставлення матриці переходів (рис. 1) та суміщеної матриці ОП (рис. 3). Результат зіставлення показаний на рис. 4 у вигляді матриці результату. Велика одиниця в комірці відповідає одиниці у відповідній комірці матриці переходів, маленька одиниця відповідає одиниці в суміщеній матриці ОП.

		a ₂	a ₅	a ₃	a ₆	a ₄	a ₀		a ₁
		0	1	2	3	4	5	6	7
a ₂	0		1	1 ₁					
a ₅	1			1	1 ₁				
a ₃	2				1	1 ₁			
a ₆	3					1	1 ₁		
a ₄	4						1 ₁	1	
a ₀	5							1	1 ₁
	6	1							1
a ₁	7	1 ₁	1 ₁						

Рисунок 4 – Матриця результату

Як можна бачити, в кожній комірці з «великою» одиницею міститься також і «маленька» одиниця. Це означає, що при обраному кодуванні станів та заданих

ОП усі мікропрограмні переходи автомата «покриваються» операціями переходів, даючи, таким чином, формальне рішення задачі алгебраїчного синтезу МПА.

Процес зіставлення матриць може бути ефективно реалізований за допомогою засобів бібліотеки Numpy.

Нехай змінна *m1* – матриця операції «+», змінна *m2* – матриця операції «+2», *mt* – матриця переходів, представлені у форматі масивів Numpy типу *ndarray*. Тоді побудова суміщеної матриці реалізується наступною командою:

```
ms = numpy.logical_or(m1,m2)
```

Зіставлення матриці переходів і суміщеної матриці ОП може бути виконане в наступний спосіб:

```
m = numpy.logical_and(mt, ms)
```

```
m = numpy.logical_xor(m, mt)
```

В результаті утворюється матриця, в якій одиничні значення відповідають переходам, непокритим жодною ОП. Кількість таких переходів можна порахувати наступною командою:

```
n = numpy.count_nonzero(m)
```

Значення *n*, рівне нулю, свідчить про знаходження формального рішення задачі алгебраїчного синтезу мікропрограмного автомата з операційним перетворенням кодів станів.

Оскільки архітектура МПА з операційним перетворенням кодів станів дозволяє зіставити будь-яку ОП будь-якому переходу автомата, запропонований в роботі підхід дозволяє за допомогою суміщеної матриці ОП абстрагуватись від конкретних операцій переходів, зосередившись на суміщенні переходів і операцій. Використання бібліотеки Numpy в якості програмного інструментарію дозволяє ефективно працювати з автоматами, кількість станів яких вимірюється сотнями і тисячами. Це дозволяє в майбутньому розробити ряд ефективних методів і алгоритмів алгебраїчного синтезу даного класу МПА та програмних реалізацій цих методів як складових спеціалізованих САПР пристроїв керування.

Список літературних джерел

6. Васильєв О. Програмування мовою Python. – Тернопіль: «Навчальна книга – Богдан», 2019. – 504 с.
7. Маккинли Уэс. Python и анализ данных / Пер. с англ. Слинкин А.А. – М.: ДМК Пресс, 2015. – 482 с.
8. Бабаков Р. М. Операционное преобразование кодов состояний в микропрограммном автомате: монография. Винница: «ТВОРИ», 2019. 208 с.
9. Бабаков Р. М. Алгебраический синтез микропрограммного автомата с операционным автоматом переходов. Информационные технологии и компьютерная инженерия. 2017. № 39. Т. 2. С. 35–41.