УДК 519.714

Д.В. Занько, ассистент

НОРМАЛИЗАЦИЯ БИНАРНОГО ОРИЕНТИРОВАННОГО ГРАФА

The paper gives a detailed introduction to the normalization procedure used in a control synthesis for the systems formalized as state-graphs. The necessary algorithms and their correctness proofs are given with short examples of the results they produce.

Рассмотрим задачу синтеза управления системой, сформулированную на ориентированном коечном графе G. Вершины в графе определяют классы состояний системы, а дуги – управления этой системой. Управления обеспечивают переходы из одного класса состояний в другой.

Каждый класс будем описывать набором элементов из {0, 1, *}:

– 0, если значение переменной состояния равно 0;
– 1, если значение переменной состояния равно 1;
– "*", если значение переменной состояния безразлично.

Все классы фиксированной размерности, равной n – числу переменных состояния системы.

Например, класс S_i = {0*1*0} включает 4 состояния:

00100
00110
01 100
01110,

которые получаем заменой разрядов, отмеченных "*", на 0 или 1. Допускаются также классы, содержащие одно состояние.

<u>Определение 1</u>. Два класса называются ортогональными, если описывающие их векторы отличаются хотя бы в одном разряде (т.е. если в одном и том же разряде эти классы имеют значения переменных состояния системы, равные 0 и 1).

<u>Определение 2</u>. Графом нормального вида будем называть граф, который представлен множеством попарно ортогональных вершин.

Для того чтобы реализовать задачу синтеза управления системой на графе, необходимо, чтобы граф был нормального вида. Это необходимо для того, чтобы в графе не было классов с одинаковыми состояниями, т.е. переход из одного класса в другой был однозначным.

Алгоритм ортогонализации двух классов можно представить следующим образом:

1. Находим любые два неортогональные класса (если имеются).
2. Из них выделяем общий класс (см. алгоритм выделения общего класса).
3. Выделяем ортогональные классы (см. алгоритм выделения ортогональных классов).

Алгоритм выделения общего класса:

Обозначим первый из двух неортогональных классов как Класс 1, второй класс – Класс 2, общий класс – Общий_Класс.

1. Выбираем первую переменную состояния системы (i= 1).
2. ЕСЛИ Класс 1[i]= 0 или Класс 2[i]= 0, ТО Общий_Класс[i]= 0 и переход на шаг 5, ИНАЧЕ переход на шаг 3.

3. ЕСЛИ Класс 1[i]= 1 или Класс 2[i]= 1, ТО Общий_Класс[i]= 1 и переход на шаг 5, ИНАЧЕ переход на шаг 4.

4. ЕСЛИ Класс 1[i]= "*" и Класс 2[i]= "*", ТО Общий_Класс[i]= "*" и переход на шаг 5.

5. Выбираем следующую переменную состояния системы (i= i + 1).

6. ЕСЛИ i > размерности класса, ТО КОНЕЦ, ИНАЧЕ переход на шаг 1.

Или проще: если значения i-х переменных совпадают, то это значение и оставляем, иначе (если не совпадают) оставляем цифру.

Пример выделения общего класса.

Допустим, найдены два неортогональные класса: 0101*** и 01**01. Общий класс будет следующим: 010101*.

Алгоритм выделения ортогональных классов.

1. Берем первый из неортогональных классов и общий класс.
2. Выбираем значащие разряды: те, которые в неортогональном классе равны "*", а в общем классе равны 0 или 1. Нумерация значащих разрядов идет справа налево.
3. Выбираем первый значащий разряд (i = 1).
4. За "уменьшаемое" берем неортогональный класс, за "вычитаемое" – общий класс, у которого все значащие разряды, кроме i-го, равны "*".
5. Из "уменьшаемого" "вычитаем" "вычитаемое" (см. алгоритм вычитания).
6. Полученную разность добавляем в список ортогональных классов.
7. Выбираем следующий значащий разряд (i = i + 1).
8. ЕСЛИ i > количества значащих разрядов, ТО переход на шаг 11, ИНАЧЕ переход на шаг 9.
9. За "уменьшаемое" берем предыдущее "вычитаемое", а за "вычитаемое" – предыдущее "вычитаемое", у которого i-й значащий разряд тот, что был заменен на "* "на шаге 4.
10. Переход на шаг 5.
11. На этом заканчивается выделение ортогональных классов для первого из неортогональных классов.

Для второго из неортогональных классов алгоритм имеет такой же вид.

Полученные два списка ортогональных классов и общий класс – совокупность ортогональных классов, полученных из двух неортогональных.

Алгоритм вычитания.

1. Выбираем первую переменную состояния системы (i= 1).
2. ЕСЛИ Уменьшаемое [i] = "*" и Вычитаемое [i] = 0, ТО Разность [i] = 1, переход на шаг 5, ИНАЧЕ переход на шаг 3.
3. ЕСЛИ Уменьшаемое [i] = "*" и Вычитаемое [i] = 1, ТО Разность [i] = 0, переход на шаг 5, ИНАЧЕ переход на шаг 4.
4. ЕСЛИ Уменьшаемое [i] = Вычитаемое [i], ТО Разность [i] = Уменьшаемое [i], переход на шаг 5.
5. Выбираем следующую переменную состояния системы (i= i + 1).
6. ЕСЛИ i > размерности класса, ТО КОНЕЦ, ИНАЧЕ переход на шаг 2.

Или проще: если значения i-х переменных совпадают, то это значение и оставляем, иначе оставляем цифру, противоположную вычитаемому.

Пример выделения ортогональных классов.

Используя предыдущий пример выделения общего класса, получим ортогональные классы. Неортогональные классы: 0101*** и 01**01*, общий класс: 010101*.

1. Берем первый из неортогональных классов (0101***) и общий класс (010101*).
2. Выбираем значащие разряды, т.е. 5-й и 6-й.
3. Выбираем первый значащий разряд (i = 1), т.е. 6-й.
4. За "уменьшаемое" берем класс 0101***, за "вычитаемое" – общий класс, у которого все значащие разряды, кроме i-го, равны "*" (0101*1*).
5. Из "уменьшаемого" "вычитаем" "вычитаемое":
6. _0101***
   0101*1*.
   0101*0*
7. Полученную разность (0101*0*) добавляем в список ортогональных классов.
8. Выбираем следующий значащий разряд (i = 2), т.е. 5-й.
9. i = количеству значащих разрядов.
10. За "уменьшаемое" берем предыдущее "вычитаемое" (0101*1*), а за "вычитаемое" – предыдущее "вычитаемое", у которого i-й значащий разряд тот, что был заменен на * на шаге 4 (010101*).
11. Из "уменьшаемого" "вычитаем" "вычитаемое":
12. _0101*1*
   010101*.
   010111*
13. Полученную разность (010111*) добавляем в список ортогональных классов.
14. Т.к. больше нет значащих разрядов, то на этом заканчивается выделение ортогональных классов для первого из неортогональных классов.

Для второго из неортогональных классов алгоритм дает следующий результат:

_01**01*	_01*101*
01*101*	010101*
01*001*	011101*

Докажем ортогональность полученных классов.

Вначале докажем, что из неортогонального класса получается совокупность ортогональных классов.

При ортогонализации класса, вычитая значение значащего разряда, получаем класс, ортогональный всем последующим ортогональным классам именно по значению этого разряда, т.к. далее получаются классы, у которых значение этого разряда противоположно тому, что записано в полученном классе.

Далее докажем, что две совокупности ортогоналяьных классов, полученных из неортогональных, ортогональны друг другу.

В общем классе (010101*) можно выделить разряды, значения которых взяты из первого неортогонального класса (4, 5) и из второго неортогонального класса (2, 3). При ортогонализации первого класса используются значения разрядов второго класса. И при получении ортогонального класса именно по значению того разряда, по которому вычитается и будет получен класс, ортогональный всем ортогональным классам, полученным из второго класса. Для второго неортогонального класса рассуждения аналогичны.

Доказательство окончено.

Докажем, что количество состояний в исходном неортогональном классе равно количеству состояний в списке ортогональных классов, полученных при ортогонализации, сложенному с количеством состояний в общем классе.

Количество состояний в исходном неортогональном классе равно 2^n, где n – количество "*" в этом классе. Количество состояний в общем классе равно 2^r, где r – количество "*" в общем классе. По способу выделения ортогональных классов видно, что в первом из выделенных ортогональных классов 2^{n-1} состояний, во втором – 2^{n-2} состояний, в третьем – 2^{n-3} состояний и т.д. и, наконец, в последнем – 2^r состояний. Т.е. необходимо доказать следующее выражение:

$2^n = 2^{n-1} + 2^{n-2} + \ldots + 2^r + 2^r$, где $0 \le r < n$, r и n – целые.

Докажем это выражение при помощи математической индукции.

Для начала докажем выражение $2^n = 2^{n-1} + 2^{n-2} + \ldots + 2^1 + 2^0 + 1$.

1. При n = 1 выражение верно:

$2^1 = 2^0 + 2^0$.

2. Допустим, что при n = k выражение верно, т.е.

$2^k = 2^{k-1} + 2^{k-2} + \ldots + 2^1 + 2^0 + 1$.

3. Докажем, что при n = k + 1 выражение также верно:

$2^{(k+1)} = 2^{(k+1)-1} + 2^{(k+1)-2} + \ldots + 2^1 + 2^0 + 1$.

После преобразования получим

$2^{(k+1)} = 2^k + 2^{k-1} + \ldots + 2^1 + 2^0 + 1$.

А сумма всех слагаемых, кроме первого, равна 2^k (п. 2). Отсюда следует, что

$2^{(k+1)} = 2^k + 2^k = 2^{(k+1)}$.

Доказательство заякончено.

Вернемся к доказательству первоначального выражения:

$2^n = 2^{n-1} + 2^{n-2} + \ldots + 2^r + 2^r$.

В силу пяредыдущего доказательства

$2^r = 2^{r-1} + 2^{r-2} + \ldots + 2^1 + 2^0 + 1$.

Следовательно,

$2^n = 2^{n-1} + 2^{n-2} + \ldots + 2^r + 2^{r-1} + 2^{r-2} + \ldots + 2^1 + 2^0 + 1$.

А это выражение верно в силу все того же предыдущего доказательства, следовательно, исходное выражение верно.

Исходя из изложенных выше доказательств (того, что количество состояний в неортогональном классе и в совокупности ортогональных классов совпадает, и того, что каждая совокупность ортогональных классов ортогональна друг другу), делаем вывод о том, что алгоритм ортогонализации работает правильно.

ЛИТЕРАТУРА

1. Герман О.В., Семерюк Д.В. Одна полиномиально разрешимая задача синтеза поведения интеллектуального робота // Автоматика и телемеханика. 2001. № 2. С. 15–24.

2. Занько Д.В. Нормализация бинарного ориентированного конечного графа // Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях: V Республиканская научная конференция студентов и аспирантов. – Гомель: ГГУ, 2002. – С. 176 – 177.