переменных необходимо применение эвристических алгоритмов, например описанных в [7].

ЗАКЛЮЧЕНИЕ

Итак, в статье проанализирована сложность описания булевых функций с помощью ДНФ, конечных автоматов и упорядоченных двоичных разрешающих диаграмм. Показано, что понятие сложности различных представлений булевых функций взаимосвязаны и дополняют друг друга.

СПИСОК ЛИТЕРАТУРЫ

1. Мельников Б.Ф. Эвристики в программировании недетерминированных игр. – «Программирование» (Известия РАН), 2001, №5, С.63–80.
2. Пивнева С.В. Моделирование задач дискретной оптимизации / С. В. Пивнева, М. А. Трифонов // Вектор науки. – № 3 (13). – Тольятти : Тольят. гос. ун-т, 2010. – С. 31–34.
3. Пивнева С.В. Программная реализация задач дискретной оптимизации / С. В. Пивнева, М. А. Трифонов // Некоторые вопросы математического моделирования дискретных систем : сб. науч. тр. – Тольятти : ТГУ, 2011. – С. 210–219.
4. Поваров Г. Конечные трансдьюсеры и недетерминированная сложность регулярного языка. Известия вузов. Математика. 2010, № 6, С. 23–31.
5. Хопкрофт Дж., Мотвани Р., Ульман Дж.. Введение в теорию автоматов, языков и вычислений. 2-е изд.: Пер. с англ. – М. : Издательский дом «Вильямс», 2002. – 528 с. : ил. – Парал. тит. англ.
6. Шуткин Ю. Асимптотически оптимальная реализация булевых функций информационными графами. Дискрет. матем., 2011, 23:4, С.80–102.
7. Мельников Б. Мультиэвристический подход к задачам дискретной оптимизации. Кибернетика и системный анализ (НАН Украины), 2006, № 3, С. 32–42.

Работа частично поддержана программой Министерства образования и науки РФ в рамках Госзадания Тольяттинского государственного университета на 2012 год (шифр 6.3072.2011) и реализации гранта федеральной целевой программы «Научные и научно-педагогические кадры инновационной России» на 2009–2013 годы (Соглашение № 14.B37.21.1934)

CALCULATION OF AUTOMATON COMPLEXITY OF BOOLEAN FUNCTIONS AS DISCRETE OPTIMIZATION PROBLEMS

N.I. Kraynyukov, candidate of pedagogical sciences, associated professor of the chair «Applied mathematics and informatics

S.V. Pivneva, candidate of pedagogical sciences, associated professor, associated professor of the chair «Higher mathematics and mathematical modeling»

Togliatti State University, Togliatti (Russia)

Keywords: Boolean functions; certainly-automatic complexity; final automatic devices.

Annotation: The problem of calculation of complexity of the description булевых functions by means of the disjunctive normal forms, the determined final automatic devices and the ordered binary diagrams is considered. Examples of realization of functions are considered by three ways, their complexity is analysed.

УДК 512.531.2

ПЕРЕБОРНЫЕ АЛГОРИТМЫ ПОДСЧЁТА ЧИСЛА КОНЕЧНЫХ ПОЛУГРУПП – ПОСТАНОВКА ЗАДАЧИ И ПРОСТЕЙШИЕ ЭВРИСТИКИ

Е.И. Кузичкина, аспирант

Тольяттинский государственный университет, Тольятти (Россия)

Ключевые слова: конечная полугруппа; алгоритмы перебора; эвристические алгоритмы.

Аннотация: Рассмотрены простые эвристические переборные алгоритмы для подсчёта числа конечных полугрупп.

ВВЕДЕНИЕ

Теория полугрупп имеет тесные связи с самыми различными математическими дисциплинами: дифференциальной геометрией, функциональным анализом, теорией графов, теорией алгоритмов, теорией формальных языков, абстрактной теорией автоматов и др. Эти связи способствуют жизненности теории полугрупп и определяют возможность её приложений. Несмотря на долгие годы изучения этого направления, для многих задач эффективных алгоритмов решения так и не было изобретено ([1]). Целью настоящей работы являлась разработка новых подходов к решению вопроса подсчёта числа конечных полугрупп на n-элементном множестве образующих.

ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ, ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

Согласно [2], ассоциативная операция – это бинарная операция ○, обладающая ассоциативностью (т.е. сочетательностью):

$$(x\circ y)\circ z = x\circ(y\circ z) \tag{1}$$

для любых элементов x, y, z. Полугруппой называют множество с заданной на нем ассоциативной бинарной операцией (S, *). Порядок множества обозначается $n \in N$ (где N = {1, 2,...}).

Задание ассоциативной операции на множестве может быть произведено различными способами. Наиболее естественным является просто перечисление всех результатов действия для любой возможной пары элементов. Такой способ задания можно представить себе в виде указания таблицы умножения, называемой также таблицей Кели.

	1	**2**
1	**2**	**1**
2	**1**	**2**

Рис. 1. Бинарная операция.

Проверка ассоциативности бинарной операции может быть осуществлена по следующей схеме. Например, возьмем множество A={1,2} с некоторой заданной операцией. Данная операция представлена в виде таблицы, изображённой на рисунке 1. Проверим на ассоциативность все возможные комбинации элементов множества А – см. рис. 2.

1 + (1 + 1) = 1 + 2 = 1
(1 + 1) + 1 = 2 + 1 = 1
Операция ассоциативна

....

2 + (2 + 2) = 2 + 2 = 2
(2+ 2) + 2 = 2 + 2 = 12
Операция ассоциативна

Рис. 2. Проверка ассоциативности.

Т.к. на всех возможных комбинациях элементов множества бинарная операция является ассоциативной, множество A = {1,2} с заданной на нём операцией является полугруппой. А если бы на какой-либо комбинации элементов ассоциативность не выполнялась, то множество A с заданной на нём операцией полугруппой не являлось бы.

Есть более «быстрые» способы проверки ассоциативности на заданной операции – например, процедура, предложенная Лайтом в 1949 г., [3]. Эту процедуру достаточно проделать лишь для каждого элемента a из некоторого порождающего множества группоида.

Покажем, как происходит поиск полугруппы на множестве на примере 2-элементного множества. Выберем 2-элементное множество *A = {1, 2}*. На таком множестве *A* могут быть определены $2^{2^2} = 16$ бинарных операций. Эти операции показаны в таблицах ниже, представленных на рис. 3, см. также [4].

Для нахождения полугрупп на множестве необходимо проверить каждую таблицу на ассоциативность, т.е. проверить является ли заданная операция ассоциативной по формуле

$$(x\circ y)\circ z = x\circ(y\circ z) \tag{1}$$

Метод проверки ассоциативности можно выбрать любой. Если выполнить проверку на всех таблицах, то обнаружится, что на множестве размером 2 (N = 2) лишь 8 таблиц являются полугруппами, см. также рис. 3.

	1	2
1	1	1
2	1	1

Полугруппа

	1	2
1	1	1
2	1	2

Полугруппа

	1	2
1	1	1
2	2	1

	1	2
1	1	1
2	2	2

Полугруппа

	1	2
1	1	
2	1	1

	1	2
1	1	2
2	1	2

Полугруппа

	1	□
1	1	2
2		□

Полугруппа

	1	2
1	1	2
2	2	

Полугруппа

	1	2
1	2	
2	1	1

	1	2
1	2	1
2	1	2

Полугруппа

	1	2
1	2	1
2	2	1

	1	2
1	2	1
2	2	2

	1	2
1	2	2
2	1	1

	1	2
1	2	2
2	1	2

	1	2
1	2	2
2	2	1

	1	2
1	2	2
2	2	2

Полугруппа

Рис. 3. Все возможные бинарные операции на 2-элементном множестве.

МЕТОДИКА ПРОВЕДЕНИЯ ВЫЧИСЛИТЕЛЬНОГО ЭКСПЕРИМЕНТА

Общая постановка задачи следующая: необходимо описать эффективные алгоритмы подсчёта количества неизоморфных полугрупп на *n*-элементном множестве.

Ниже представлена таблица в первой строке, которой представлена информация о размерности полугруппы, во второй строке количество всех возможных операций на множестве, в третьей строке количество полугрупп на множестве и в четвертой частота появления полугрупп.

Таблица 1. Анализ полугрупп.

N	1	2	3	4	5	6	7
Количество претендентов на полугруппу	1	16	19 683	$4{,}29 \cdot 10^{9}$	$2{,}98 \cdot 10^{17}$	$1{,}03 \cdot 10^{28}$	$2{,}57 \cdot 10^{41}$
Количество полугрупп	1	8	113	3 492	183 732	17 061 118	7 743 056 064
Отношение	100%	50%	0,6%	$8{,}1 \cdot 10^{-5}$%	$6{,}1 \cdot 10^{-11}$%	$1{,}6 \cdot 10^{-19}$%	$3{,}0 \cdot 10^{-30}$%

Отсюда понятно, что задача поиска количества полугрупп на больших размерностях становится нетривиальной.

Метод 1 – полный перебор.

На компьютере 2.00 – GHz – Intel T-5870 – 2 GB RAM за реальное для ожидания время можно решить эту задачу:

- на множестве из 3 элементов – за 15 минут;
- из 4 элементов – за 10 часов.

С точки зрения асимптотической оценки сложности алгоритма имеем:

$$f(n) \in O(n^{n^2} \cdot n^3) \qquad (2)$$

Здесь n^{n^2} – число, получающееся при переборе всех возможных операций на n-элементном множестве, а n^3 – при проверке ассоциативности каждой операции.

Отсюда понятно, что для подсчёта полугрупп в более высоких порядках множества необходимо применять специальные эвристики.

Метод 2 – авторский, мультиэвристический.

В данной задаче были использованы ряд эвристик, которые позволили сократить время выполнения программы в разы. Далее они будут подробно рассмотрены. Многие из этих эвристик согласованы с подходом, предложенным в [5].

Эвристика 1. Не нужно перебирать все возможные операции на множестве, нужно строить полугруппу *по мере заполнения* таблицы бинарной операции. Эта эвристика основана на наблюдении, представленном на рисунке 4.

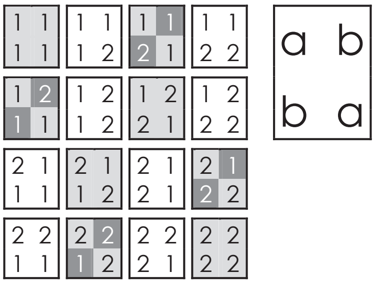

Рис. 4. Некоторые связи между элементами полугрупп.

Пусть есть 2-элементное множество *A={1,2}*. Выпишем все возможные бинарные операции на этом множестве. При этом обратим внимание на следующий факт. Если элемент, стоящий на пересечении первого столбца и первой строки *(1,1)* равен *a*, и элемент, стоящий на пересечении второго столбца и второй строки *(2,2)*, также равен *a* (где *a* – некоторый элемент из множества *A*), то элемент, стоящий на пересечении первой строки и второго столбца *(1,2)* должен быть равен элементу на пересечении второй строки и первого столбца *(2,1)*.

Таким образом, становится ясно, что существуют связи в полугруппе между элементами (наборами элементов), при которых полугруппу построить невозможно. Т.е. необходимо проверять возможность построения полугруппы на данном наборе, отсекая варианты построения «изначально неверных» таблиц. Рассмотрим пример для бинарной операции, представленной на рис. 5.

	1	2	3	...
1	1	1	?	?
2	2	1	?	?
3	?	?	?	?
...	?	?	?	?

Рис. 5. Набор, на котором нельзя построить полугруппу.

Эту таблицу не имеет смысла дальше заполнять, какими либо вариантами – поскольку всегда найдется тройка, на которой свойство ассоциативности не выполнится.

Эвристика 2. Вторая эвристика используется для осуществления первой. Заполнять таблицу можно по-разному, и способ заполнения таблицы влияет на порядок нахождения «тупиковых» расстановок в таблице. Например, таблицу заполнять можно последовательно, с начала первая строка, потом вторая и т.д.:

1	n+1	n+2	...	2n-1
2n	2	3n-1	...	4n-3
2n+1	4n-2	3	...	6n-5
...	...	...	...	...
3n-2	5n-4	7n-6	...	n

Рис. 6. Способ заполнения полугруппы.

Аналогично можно заполнять последовательно по столбцам. По проведенным исследованиям, лучшим порядком заполнения является способ, представленный на рисунке 6.

Эвристика 3. Т.к. необходимо постоянно проверять различные элементы на ассоциативность, нужно рассмотреть этапы вычисления одной операции и возможности сокращения действий, производимых компьютером. Проверка операции ассоциативности осуществляется в 5 этапов (рис. 7.).

(a * b) * c = a * (b * c)
(a * b) этап 1
(b * c) этап 2
(a * b) * c этап 3
a * (b * c) этап 4
(a * b) * c ? a * (b * c) этап 5

Рис. 7. Этапы вычисления ассоциативной операции.

Для ускорения вычислений 1-го и 2-го этапов, а в последующем и 3-го и 4-го этапов, можно воспользоваться адресными ссылками. Таким образом, при построении таблицы, вычисление 1-го и 2-го этапов будет осуществляться «автоматически». При заполнении таблицы ссылки будут указывать на вычисленные элементы.

Этапы 3 и 4 имеет зависимость от вычисления этапов 1 и 2. Здесь необходимо прописывать адрес для каждого вновь заполняемого элемента. Например, для заполняемого элемента (a, b) необходимо настроить ссылки для:

1 * (a * b) (a * b) * 1
2 * (a * b) (a * b) * 2
... ...
n * (a * b) (a * b) * n

Эвристика 4. Для её описания необходимо представить все объекты программы (алгоритма); они представлены на рис. 8. Таблица бинарной операции – объект для хранения состояния текущей бинарной операции. Список троек для проверки ассоциативности бинарной операции – N^3 вариантов различных операций, необходимых для выяснения является ли группоид полугруппой. Каждая тройка содержит в себе атрибуты, отображающие 5 этапов вычисления проверки ассоциативности (см. эвристику 3), и статус (подробнее см. эвристику 5). Стек операций – список выполненных действий по заполнению таблицы бинарной операции.

Рис. 8. Объекты программы

Эвристика 4 – это ведение стека совершаемых действий. Так как построение перебора вариантов бинарных операций на множестве идёт не линейно, а по оптимизированной схеме (см. эвристику 2), используя статусные показатели троек (см. эвристику 5), возращение назад по дереву *не* является программируемым – поэтому его необходимо запоминать. Стек совершаемых операций хранит в себе информацию о заполняемой клетке в таблице бинарной операции, а также значение заполняемого элемента.

Эвристика 5. Данная эвристика – это хранение статуса у каждой тройки для проверки ассоциативности бинарной операции.

Есть несколько статусов:

1) Если $a * (b * c) = (a * b) * c$, то статус равен 4 или OK. Это означает, что операция выполнена, следовательно ассоциативность есть.

2) Если элемент *(a, b)* вычислен (т.е. уже установлен), элемент *(b, c)* вычислен, элемент *(a, (b, c))* вычислен, а элемент *((a, b), c)* не вычислен, то статус будет равен 1. Это означает, что мы можем однозначно определить элемент *((a,b), c)*. Он будет равен элементу *(a, (b,c))*.

Например *(a, b) = a*, *(b, c)= b*, тогда *(a, (b,c)) = a*, *((a,b), c) = ?*.

	a	b	c
a		a	?
b			b
c			

Рис. 9. Набор элементов полугруппы.

На данном наборе (рис. 9) элемент *(a,c)* не может быть определен, каким-либо элементом помимо элемента *a*. Таким образом, статус 1 показывает все точки, в которых существует однозначное действие. Если этот статус определен, то, нарушая подход из эвристики 2, следующим элементом для заполнения должен стать элемент *((a,b), c)*, с пометкой в стеке «сразу для удаления». Это означает, что на место этого элемента в матрице бинарной операции не может быть установлен никакой другой элемент, поэтому при подъёме по дереву перебора бинарных операций его необходимо сразу удалять (а не изменять следующим по порядку).

3) Если элемент *(a, (b,c))* не вычислен, элемент *((a,b), c)* не вычислен, элемент *(b,c)* вычислим, элемент *(a,b)* вычислен, то статус операции будет равен 2. Это означает, что заполнять элементы *(a, (b,c))* и *((a,b), c)* необходимо вместе, потому что ничто иное не имеет смысла – операция ассоциативности не выполнится. Например: *(a, b) = c*, *(b, c) = a*, тогда *(a, (b,c)) = ?*, *((a,b), c) = ?*.

	a	b	c
a	?	c	
b			a
c			?

Рис. 10. Набор элементов полугруппы.

В данном случае (рис. 10) операция, также как и в предыдущий раз, становится выше по приоритету и должна быть выполнена следующей по очереди. В стек заносится элемент *(a, (b,c))* и значение элемента, устанавливаемого в матрицу бинарной операции, а также ставится пометка о том, что вместе с этим элементов был заполнен элемент *((a,b), c)* тем же значением.

4) Если элемент *(a, (b,c))* не равен *((a,b), c)*, то статус равен 3. Это означает, что ассоциативность на этой тройке не достигнута. Дальнейшего построение полугруппы на этой матрице не имеет смысла, так как имеется невыполненная операция. Здесь имеет смысл только откат по стеку совершённых действий.

Эвристика 6. Для вычисления 5-го этапа троек для проверки ассоциативности не обязательно на каждом шаге проверять обновилась ли информация для вычисления статуса у тройки или нет. Достаточно проверить тройки, в которых присутствует только что изменённый элемент.

Например, если изменен элемент *(a,b)*, то необходимо проверить все тройки, в которых есть:

- $(a * b) * c$;
- $c * (a * b)$;
- $(c * d) = a$, тогда нужно проверить тройку $(c * d) * b$;
- $(c * d) = b$, тогда нужно проверить тройку $a * (c * d)$.

С первым и вторым пунктами всё просто. Для быстрого поиска третьего и четвертого пунктов можно усложнить структуру объекта 2 – для того, чтобы можно быстро осуществлять доступ к таким элементам; можно также усложнить структуру матрицы бинарной операции – для быстрого определения нужных элементов.

Полный алгоритм решения задачи 1.

Используя все выше перечисленные эвристики, строится алгоритм:

1. В соответствии с эвристикой 2 начинаем заполнять матрицу бинарной операции.
2. После каждого заполненного элемента:
   2.1. Записываем в стек совершенное действие (см. эвристику 4).
   2.2. Настраиваем ссылки на элементы для вычисления третьего и четвертого этапа для проверки ассоциативности операции (см. эвристику 3).
   2.3. Проверяем все необходимые тройки для вычисления 5-го этапа и смены статуса тройки (см. эвристику 6).
3. Проверяем, есть ли не выполненные операции (ищем тройки со статусом 3):
   3.1. Есть. Делаем откат по дереву совершённых операций. И снова возвращаемся на пункт 3.
   3.2. Нет. Переходим к пункту 4.
4. Проверяем, есть ли «приоритетные» элементы для заполнения матрицы (см. эвристику 5).
   4.1. Есть. Заполняем в матрице «приоритетный» элемент. Переходим к пункту 2.
   4.2. Нет. Переходим к пункту 1.

НЕКОТОРЫЕ ПОЛУЧЕННЫЕ РЕЗУЛЬТАТЫ

Применённые технологии описаны в таблице 2, полученные автором результаты приведены в таблице 3.

Таблица 2. Применяемые технологии для поиска решения.

	Машина для тестирования:	Язык программирования:
Andreas Distler	2.66 GHz Intel X-5430 16 GB RAM	64-bit executable of Minion version 0.9
Кузичкина Елена	2.00 GHz Intel T-5870 2 GB RAM	VSC++

Таблица 3. Результаты работы алгоритмов.

n	1	2	3	4	5	6	7
Количество полугрупп	1	8	113	3 492	183 732	17 061 118	7 743 056 064
Время выполнения алгоритма №1, не представленного в этой работе	ξ	ξ	ξ	ξ	26s	10021s	–
Время выполнения алгоритма №2, представленного в данной работе	ξ	ξ	ξ	ξ	9s	1930s	449 150s
Время выполнения Andreas'a Distler'a	ξ	ξ	ξ	ξ	4s	334s	73 563s

КРАТКОЕ ОБСУЖДЕНИЕ РЕЗУЛЬТАТОВ

В настоящее время нет возможностей точного сравнения результатов, полученных автором и приведённых в [6], – поскольку в двух случаях использовались различные вычислительные ресурсы. Однако в планах автора настоящей работы – продолжение исследований, улучшение полученных результатов, точное сравнение обоих подходов, создание распределённых версий описанных алгоритмов, а в перспективе – существенное улучшение результатов Andreas'a Distler'a.

Работа автора частично поддержана региональным грантом РФФИ № 13-01-97003, а также частично поддержана Федеральной целевой программой "Научные и научно-педагогические кадры инновационной России" на 2009–2013 годы, соглашение № 14.В37.21.1934.

СПИСОК ЛИТЕРАТУРЫ

1. Шеврин Л.Н. Что такое полугруппа // Соросовский Образовательный Журнал. 1997. № 4. С. 99–104.
2. Полугруппа [Электронный ресурс]: Материал из «Викия-сеть»: Версия 3, сохранённая в 17:17 UTC 2 сентября 2010 / Авторы Викия-сеть// Викия-сеть. – Режим доступа: http://ru.math.wikia.com/wiki/Полугруппа
3. Клиффорд А., Престон Г. Алгебраическая теория полугрупп. – М.: Мир, 1972. – 286 с.
4. Semigroup with two elements [Электронный ресурс]: Материал из Википедии – свободной энциклопедии: Версия 488388842, сохранённая в 20:24 UTC 20 апреля 2012 / Авторы Википедии // Википедия, свободная энциклопедия. – Электрон. дан. – Сан-Франциско: Фонд Викимедиа, 2012. – Режим доступа: http://en.wikipedia.org/w/index.php?title=Semigroup_with_two_elements&oldid=488388842
5. Мельников Б. Мультиэвристический подход к задачам дискретной оптимизации. – Кибернетика и системный анализ (НАН Украины), 2006, № 3, С. 32–42.
6. Classification and enumeration of finite semigroups [Электронный ресурс]: Research@StAndrews Full Text: Версия: 10023/945, сохраненная в июне 2010 / Andreas Distler // Research@StAndrews Full Text – Режим доступа: http://hdl.handle.net/10023/945

ALGORITHMS FOR CALCULATING THE NUMBER OF FINITE SEMIGROUPS: PROBLEM STATEMENT AND SIMPLE HEURISTICS

E.I. Kuzichkina, postgraduate student
Togliatti state university, Togliatti (Russia)

Keywords: finite semigroup; exhaustive search; heuristics.

Annotation: Simple heuristic exhaustive search for calculation of the number of finite semigroups were obtained.

УДК 512.531.2

ПАРАЛЛЕЛЬНАЯ РЕАЛИЗАЦИЯ НЕКОТОРЫХ АЛГОРИТМОВ ПОДСЧЁТА ЧИСЛА КОНЕЧНЫХ ПОЛУГРУПП

Е.И. Кузичкина, аспирант
Д.И. Власов, студент
Тольяттинский государственный университет, Тольятти (Россия)

Ключевые слова: конечная полугруппа; алгоритмы перебора; эвристические алгоритмы, распределенные вычисления, параллельные алгоритмы.

Аннотация: Рассмотрен эвристический переборный алгоритм для подсчёта числа конечных полугрупп и вариант его параллелизации.