Министерство образования и науки Российской Федерации

Федеральное агентство по образованию

Федеральное государственное образовательное учреждение высшего профессионального образования «Санкт-Петербургский государственный университет»

Математико-механический факультет

Кафедра исследования операций

Дипломная работа

Исаев Глеб Андреевич

Поиск булевых функций максимальной нелинейности при наличии ограничений

Заведующий кафедрой:
д. ф.-м. н., профессор И. В. Романовский
Научный руководитель:
к. ф.-м. н., доцент И. В. Агафонова
Рецензент:
к. ф.-м. н., доцент О. М. Дмитриева

Санкт-Петербург
2016 г.

Saint Petersburg State University
Faculty of Mathematics and Mechanics
Department of Operations Research

Graduation Thesis

Isaev Gleb Andreevich

Finding Boolean functions with maximal nonlinearity under constraints

Head of Department:
Professor I. V. Romanovsky
Scientific Supervisor:
Associate Professor I. V. Agafonova
Reviewer:
Associate Professor O. M. Dmitrieva

Saint Petersburg
2016

Содержание

Введение

Работа посвящена построению и поиску булевых функций максимальной нелинейности при наличии криптографически важных ограничений, таких, как сбалансированность, корреляционная иммунность и устойчивость. Мы посчитаем максимальную нелинейность при ограниченной устойчивости в определенных случаях и рассмотрим некоторые методы поиска и построения булевых функций с высокой нелинейностью и некоторой фиксированной устойчивостью.

В данной работе отметим три основных направления. Во-первых, для булевой функции от n переменных мы будем максимизировать нелинейность при ограниченной устойчивости в определенных случаях, воспользовавшись линейными ограничениями и некоторыми теоремами о границах нелинейности. Во-вторых, мы изучим метод построения устойчивых функций с хорошей нелинейностью, который был представлен в [2]. В-третьих, мы рассмотрим приближенные алгоритмы поиска булевых функций, такие, как генетический алгоритм и алгоритм направленного поиска. Они предназначены для нахождения булевых функций с хорошими свойствами, таких, как нелинейность, высокий порядок корреляционной иммунности и т.д. Такой алгоритм был рассмотрен в [1].

Работа организована следующим образом. В первом разделе даны основные определения и обозначения, применяемые в данной работе. Во втором разделе приводятся теоремы, связанные с доказательством существования определенных границ нелинейности, вместе с примерами. В третьем разделе рассматривается метод построения устойчивых булевых функций с высокой нелинейностью, который был представлен в [2]. В четвертом разделе описывается использование целочисленного программирования для максимизации нелинейности булевых функций с ограниченной устойчивостью. В пятом разделе приводятся приближенные алгоритмы поиска булевых функций, такие, как генетический алгоритм и алгоритм направленного поиска булевых функций с хорошими свойствами. Автор работы запрограммировал методы, описанные в двух последних разделах, на языках MATLAB и PASCAL. В MATLAB были использованы встроенные функции целочисленного программирования и генетического алгоритма.

1 Основные определения и обозначения

Булева функция — это такая функция f от n переменных, у которой каждый независимый аргумент $x_i, i \in 1:n$ и каждое значение функции $f(x_1, \ldots, x_n)$ равно 0 или 1. Обычно булеву функцию отождествляют с её вектором значений. Множество векторов размерности n, состоящих из логических переменных, обозначим через V_n, а множество булевых функций от n переменных — через $\mathcal{F}_n$.

Функция от n переменных имеет 2^n значений. Так как на каждом векторе булева функция может принимать значение либо 0, либо 1, то количество всех n-арных булевых функций равно 2^{2^n}.

Любую булеву функцию можно единственным образом записать в ***алгебраической нормальной форме (АНФ)***, то есть

$$f(x_1, x_2, \ldots, x_n) = a_0 \oplus a_1 x_1 \oplus a_2 x_2 \oplus \cdots \oplus a_n x_n \oplus a_{12} x_1 x_2 \oplus a_{13} x_1 x_3 \oplus \cdots \oplus a_{1\ldots n} x_1 \ldots x_n,$$

где $a_{i_1 \ldots i_n} \in \{0, 1\}$, а операциями в этом многочлене служат конъюнкция и сложение по модулю 2 (обозначается знаком $\oplus$).

Алгебраическая степень функции — это степень АНФ этой функции, то есть максимальная степень среди степеней всех мономов. Функции, имеющие степень не выше 1, называются *аффинными.*

Введем скалярное прозведение $\langle a, b \rangle = a_1 b_1 \oplus \cdots \oplus a_n b_n$. ***Преобразованием Уолша–Адамара*** булевой функции f от n переменных называется целочисленная функция W_f, заданная на множестве V_n равенством

$$W_f(\omega) = \sum_{u \in V_n} (-1)^{\langle u, \omega \rangle \oplus f(u)}.$$

Преобразованием Фурье булевой функции f от n переменных называется целочисленная функция F_f, заданная на множестве V_n равенством

$$F_f(\omega) = \sum_{u \in V_n} (-1)^{\langle u, \omega \rangle} f(u).$$

Широко известна формула обращения преобразования Фурье в преобразование Уолша–Адамара:

$$W_f(\omega) = \delta_0(\omega) \cdot 2^n - 2F_f(\omega),$$

где

$$\delta_0(\omega) = \begin{cases} 1, & \omega = 0; \\ 0, & \omega \neq 0. \end{cases}$$

Вес Хэмминга $wt(f)$ булевой функции f — это число единиц вектора значений функции f.

Расстоянием Хэмминга между двумя функциями f и g называется вес функции $f \oplus g$ или, иначе говоря, число таких векторов x, при которых $f(x) \neq g(x)$.

Нелинейность N_f булевой функции f — это расстояние Хэмминга между f и множеством аффинных функций.

Широко известна формула подсчета нелинейности:

$$N_f = 2^{n-1} - \frac{1}{2} \cdot \max_{\omega \in V_n} \big(|W_f(\omega)|\big).$$

Рассмотрим важнейшие классы булевых функций.

Бент-функцией называется такая булева функция от n переменных (n четно), что модуль каждого коэффициента Уолша-Адамара этой функции равен $2^{\frac{n}{2}}$.

Булева функция называется ***сбалансированной***, если в таблице истинности количество нулей совпадает с количеством единиц.

Функция $f(x_1, x_2, \ldots, x_n)$ называется ***корреляционно иммунной*** порядка m, если при фиксации не более m координат вероятность распределения значений любого из сужений функции не меняется. Эквивалентное определение: функция $f(x_1, x_2, \ldots, x_n)$ называется ***корреляционно иммунной*** порядка m, если её преобразование Уолша–Адамара W_f удовлетворяет равенству $W_f(\omega) = 0$ при $1 \leqslant wt(\omega) \leqslant m$. Очевидно, что если функция корреляционно иммунна порядка $m, m > 1$, то она корррреляционно иммунна порядка $j, 1 \leqslant j < m$. Введем следующее обозначение:

$$corf = \max\{m \in \mathbb{N} \mid f \text{ — корреляционно иммунна порядка } m\}.$$

Функция $f(x_1, x_2, \ldots, x_n)$ называется m***-устойчивой***, если при фиксации не более m координат любое из сужений функции f сбалансировано. Эквивалентное определение: функция $f(x_1, x_2, \ldots, x_n)$ называется m***-устойчивой***, если её преобразование Уолша–Адамара W_f удовлетворяет равенству $W_f(\omega) = 0$ при $0 \leqslant wt(\omega) \leqslant m$. Очевидно, что если функция m-устойчива, $m > 1$, то она j-устойчива, где $1 \leqslant j < m$.

Заметим, что если функция является корреляционно иммунной порядка m, то она может быть и m-устойчивой, если дополнительно проверить условие $W_f(\omega) = 0$, где $wt(\omega) = 0$. Для этого достаточно показать, сбалансирована ли наша функция или нет.

Введём ещё одно обозначение:

$$sut\ f = \max\{d \in \mathbb{N} \mid f \text{ является } d - \text{устойчивой функцией}\}.$$

Заметим, что для любой $f \in \mathcal{F}_n$ справедливо неравенство $n - 1 \geqslant sut\ f$.

Неравенства Зигенталера:

- Если f — корреляционно иммунная функция порядка m от n переменных, то $deg\ f \leqslant n - m$.
- Если f является m-устойчивой функцией и $m \leqslant n - 2$, то $deg\ f \leqslant n - m - 1$.

Эти неравенства приведены вместе с доказательством в [6].

2 Некоторые теоремы и леммы, связанные с нелинейностями

Использование устойчивых функций в криптографических целях вызывает интерес к изучению нелинейности этих функций. Следующий раздел посвящен оценкам нелинейности корреляционно иммунных и устойчивых функций, а также условиям достижимости этих оценок. Все теоремы и утверждения были взяты из [4].

Теорема 1. *Справедливы следующие утверждения:*
1) Если $f \in \mathcal{F}_n, cor\ f = m \leqslant n-1,$ *то* $N_f \leqslant 2^{n-1} - 2^m$;
2) Если $f \in \mathcal{F}_n, sut\ f = m \leqslant n-2,$ *то* $N_f \leqslant 2^{n-1} - 2^{m+1}$.

Пример 1. Достижимость оценки 1 из теоремы 1.

Пусть $n = 6, m = 3$. Возьмем функцию

$$f(x_1, \ldots, x_6) = \bigoplus_{1 \leqslant i < j < k \leqslant 6} x_i x_j x_k \oplus \bigoplus_{i=1}^{4} x_i x_{i+1} \oplus x_1 x_5 \oplus \bigoplus_{i=1}^{5} \oplus 1.$$

Такая функция является несбалансированной корреляционно иммунной порядка 3 функцией. Её нелинейность равна 24: $N_f = 24$, т.е. оценка достигается.

Пример 2. Достижимость оценки 2 из теоремы 1.

Пусть $n = 3, m = 0$. Возьмем функцию

$$f(x_1, x_2, x_3) = 1 \oplus x_2 \oplus x_3 \oplus x_1 x_2 \oplus x_2 x_3.$$

Такая функция является сбалансированной. Её нелинейность равна 2: $N_f = 2$, т.е. оценка достигается.

Теорема 2. *Пусть* $f \in \mathcal{F}_n, sut\ f = m \leqslant n-3, deg\ f + sut\ f \leqslant n-2$. *Тогда* $N_f \leqslant$ $\leqslant 2^{n-1} - 2^{m+2}$.

Пример 3. Достижимость оценки из теоремы 2.

Пусть $n = 4, m = 0$. Возьмем функцию $f = B217$[1].

Такая функция является сбалансированной. Её нелинейность равна 4: $N_f = 4$, т.е. оценка достигается.

Следствие 1. *Пусть* $f \in \mathcal{F}_n, f \neq const, sut\ f = m \leqslant n-2, N_f = 2^{n-1} - 2^{m+1}$. *Тогда* $deg\ f + sut\ f = n-1$.

Функция из примера 2 как раз удовлетворяет этому следствию.

Теорема 3. *Пусть* $f \in \mathcal{F}_n$.

1. *Если* $cor\ f = m \leqslant \frac{n}{2} - 1,$ *то*

$$N_f \leqslant 2^{n-1} - 2^{\frac{n}{2}-1} - 2^m.$$

[1]Здесь и далее вектор значений булевой функции от большого числа переменных будем писать в шестнадцатеричном виде.

2. Если sut $f = m \leqslant \frac{n}{2} - 2$, *то*

$$N_f \leqslant 2^{n-1} - 2^{\frac{n}{2}-1} - 2^{m+1}.$$

Пример 4. Достижимость оценки 1 из теоремы 3.

Пусть $n = 4, m = 1$. Возьмем функцию $f = 6DDE$.

Такая функция является несбалансированной корреляционно иммунной порядка 1 функцией. Её нелинейность равна 4: $N_f = 4$, т.е. оценка достигается.

Пример 5. Достижимость оценки 2 из теоремы 3.

Пусть $n = 6, m = 1$. Возьмем функцию $f = BF6882394A54A57D$.

Такая функция является 1-устойчивой. Её нелинейность равна 24: $N_f = 24$, т.е. оценка достигается.

Теорема 4. *Пусть* $f \in \mathcal{F}_n, sut\ f = m \leqslant \frac{n}{2} - 2$. *Тогда*

$$N_f \leqslant 2^{n-1} - 2^{m+1} \left\lceil \frac{2^{n-m-2}}{\sqrt{2^n - \sum_{i=1}^{m} \binom{n}{i}}} \right\rceil.$$

Функция из примера 5 достигает оценки из теоремы 4.

Обозначим через $N_{max}(n, m)$ максимально возможную нелинейность m-устойчивой булевой функции из $\mathcal{F}_n$. Любую произвольную функцию от n переменных назовем (-1)-устойчивой функцией, поскольку у неё не существует подфункций от $n+1$ переменных, а любую сбалансированную функцию — 0-устойчивой функцией. Для (-1)-устойчивых функций справедливы следующие утверждения:

- $N_{max}(n, -1) \leqslant 2^{n-1} - 2^{\frac{n}{2}-1}$ (достигается при бент-функциях);
- При нечетных $n \leqslant 7$: $N_{max}(n, -1) = 2^{n-1} - 2^{\frac{n-1}{2}}$;
- При нечетных $n \geqslant 15$: $N_{max}(n, -1) > 2^{n-1} - 2^{\frac{n-1}{2}}$.

Для 0-устойчивых функций $f \in \mathcal{F}_n$ имеем:

- $N_f < 2^{n-1} - 2^{\frac{n}{2}-1}$;
- $N_{max}(n, m) < 2^{n-1} - 2^{\frac{n}{2}-1}$ при $m \geqslant 0$.

Для m-устойчивых функций $f \in \mathcal{F}_n$ известны следующие факты:

- Если $m \geqslant n - 2$, то $deg\ f \leqslant 1$ и $N_{max}(n, m) = 0$;
- $N_{max}(n, n-3) = 2^{n-2}$;
- $N_{max}(n, n-4) = 2^{n-1} - 2^{n-3}$.

В частности известно, что

- $N_{max}(4, 0) = 4$;
- $N_{max}(5, -1) = N_{max}(5, 0) = N_{max}(5, 1) = 12$;

- $N_{max}(6,0) = 26$;
- $N_{max}(6,1) = N_{max}(6,2) = 24$;
- $N_{max}(7,-1) = N_{max}(7,0) = N_{max}(7,1) = 56$.

Теорема 5. *Пусть* $\frac{2n-7}{3} \leqslant m \leqslant n-2$. *Тогда выполняется* $N_{max}(n,m) = 2^{n-1} - 2^{m+1}$.

Позднее было доказано, что эта нелинейность также достигается при ограничениях $\frac{2n-8}{3} \leqslant m \leqslant n-2$.

Пример 6. Достижимость максимальной нелинейности из теоремы 5.

Пусть $n = 4, m = 1$. Возьмем функцию $f = D18B$.

Такая функция является 1-устойчивой. Её нелинейность равна 4: $N_f = 4$, т.е. максимальная нелинейность достигается.

Теорема 6. *Для целых* m *и* n, *удовлетворяющих неравенствам* $\frac{2n-7}{3} \leqslant m \leqslant n - \log_2 \frac{n+2}{3} - 2$, *существует* m*-устойчивая булева функция из* $\mathcal{F}_n$ *с нелинейностью* $2^{n-1} - 2^{m+1}$, *удовлетворяющая неравенству Зигенталера для каждой отдельной переменной.*

Функция из примера 6 удовлетворяет условиям теоремы 6 и является искомой функцией.

3 Метод построения устойчивых функций порядка m с высокой нелинейностью

В данном разделе опишем метод построения устойчивой функции порядка m от n переменных при определенном l_1 с нелинейностью $N_f = 2^{n-1} - 2^{l_1-1}$, который был представлен в [2]. Особенно важно отметить, что такая нелинейность очень высока, но не всегда максимальна.

Вход. n ($n \geqslant 4$; количество аргументов булевой функции), m ($1 \leqslant m \leqslant n-3$; устойчивость).

1. Пусть $l_1,\ m+1 \leqslant l_1 \leqslant n$ – такое наименьшее число, удовлетворяющее
$$\binom{l_1}{m+1} + \binom{l_1}{m+2} + \cdots + \binom{l_1}{l_1} \geqslant 2^{n-l_1}.$$

2. Выберем 2^{n-l_1} векторов $A_0, A_1, \ldots, A_{2^{n-l_1}-1}$ из множества V_{l_1} с весом не меньшим, чем $m+1$.

3. Определим функцию $f \in \mathcal{F}_n$ следующим образом:
$$f(y_1, \ldots, y_{n-l_1}, x_1, \ldots, x_{l_1}) = f(y, x) = A_y \cdot x.$$

Функция, определенная в шаге 3, получается m-устойчивой, и она имеет нелинейность $N_f = 2^{n-1} - 2^{l_1-1}$. Это объясняется теоремами, приведенными в [2]. Алгебраическая степень такой функции $f(x)$ выводится из равенства $deg(f) = n - l_1 + 1$.

Этот метод и связанные с ним факты рассматриваются в [2].

Пример. Построим 1-устойчивую нелинейную булеву функцию от 7 переменных.

Вход: $n = 7, k = 1$.

1. Из-за того, что $\binom{3}{2} + \binom{3}{3} \ngeqslant 2^{7-3}$ и $\binom{4}{2} + \binom{4}{3} + \binom{4}{4} \geqslant 2^{7-4}$, мы имеем $l_1 = 4$.

2. Выберем 8 векторов $A_i \in V_4$ с весом $wt(A_i) \geqslant 2$.
$$A_0 = (1,1,0,0), A_1 = (1,0,1,0),$$
$$A_2 = (1,0,0,1), A_3 = (0,1,1,0),$$
$$A_4 = (0,1,0,1), A_5 = (0,0,1,1),$$
$$A_6 = (1,1,1,0), A_7 = (1,1,0,1).$$

3. Определим функцию $f : V_7 \to V_1$ таким образом:
$$f(y,x) = f(y_1, y_2, y_3, x_1, x_2, x_3, x_4) = A_y \cdot x =$$
$$= (y_1 \oplus 1)(y_2 \oplus 1)(y_3 \oplus 1)(x_1 \oplus x_2) \oplus (y_1 \oplus 1)(y_2 \oplus 1)y_3(x_1 \oplus x_3) \oplus$$
$$\oplus (y_1 \oplus 1)y_2(y_3 \oplus 1)(x_1 \oplus x_4) \oplus (y_1 \oplus 1)y_2y_3(x_2 \oplus x_3) \oplus$$
$$\oplus y_1(y_2 \oplus 1)(y_3 \oplus 1)(x_2 \oplus x_4) \oplus y_1(y_2 \oplus 1)y_3(x_3 \oplus x_4) \oplus$$
$$\oplus y_1y_2(y_3 \oplus 1)(x_1 \oplus x_2 \oplus x_3) \oplus y_1y_2y_3(x_1 \oplus x_2 \oplus x_4).$$

Такая функция f 1-устойчива и имеет нелинейность $N_f = 2^6 - 2^3 = 56$ и степень $deg\ f = 7 - 4 + 1 = 4$.

Пазалик и Йоханссон с помощью этого метода смогли построить функции с высокой нелинейностью и привели таблицу нелинейностей m-устойчивых функций от n переменных.

	n					
m	5	6	7	8	9	10
0	12	24	56	112	240	480
1	12	24	56	112	240	480
2	8	24	48	112	240	480
3	0	16	48	96	224	480
4	0	0	32	96	192	448
5	0	0	0	64	192	448
6	0	0	0	0	128	384

Таблица 1. Нелинейность N_f, полученная в соответствии с методом построения из [2].

Стоит заметить, что из неравенства Зигенталера следует, что $m \leqslant n - 1 - deg\ f$. Поэтому функция, имеющая устойчивость порядка не менее, чем $n - 2$, является либо аффинной, либо нулевой, что и объясняет нулевую нелинейность.

4 Использование целочисленного программирования для максимизации нелинейности при ограниченной устойчивости

Чтобы максимизировать нелинейность при ограниченной устойчивости, воспользуемся линейными ограничениями, которые будут приведены ниже,

Проблема существования m-устойчивой булевой функции $f(x)$ с нелинейностью N_f эквивалентна задаче нахождения такой булевой функции $f(x)$, преобразование Фурье которой удовлетворяет набору следующих ограничений:

$$\begin{gathered} F_f(0) = 2^{n-1},\ \text{сбалансированность}, \\ F_f(\omega) = 0,\ 1 \leqslant wt(\omega) \leqslant m,\ m - \text{устойчивость}, \\ |F_f(\omega)| \leqslant 2^{n-1} - N_f,\ \omega \neq 0,\ \text{нелинейность}. \end{gathered} \tag{1}$$

Запишем экстремальную задачу на максимизацию нелинейности, фиксируя порядок устойчивости:

$$N_f \to sup$$

при условии, что

$$\begin{gathered} F_f(0) = 2^{n-1}, \\ F_f(\omega) = 0,\ 1 \leqslant wt(\omega) \leqslant m, \\ F_f(\omega) + N_f \leqslant 2^{n-1},\ \omega \neq 0, \\ -F_f(\omega) + N_f \leqslant 2^{n-1},\ \omega \neq 0, \\ N_f \geqslant 0, N_f - \text{целое}. \end{gathered} \tag{2}$$

Мы знаем по определению, что $F_f(\omega) = A_\omega f$, где A — это матрица, состоящая из ± 1, которые получены из функции Уолша $(-1)^{\langle \omega, x \rangle}$ (матрица Уолша–Адамара), A_ω — это строка матрицы Уолша–Адамара A, соответствующая ω (см. [3]). Тогда можно записать задачу 2 как задачу целочисленного программирования:

$$N_f \to sup$$

при условии, что

$$\begin{gathered} A_{\mathbb{O}} f = 2^{n-1}, \\ A_\omega f = 0,\ 1 \leqslant wt(\omega) \leqslant m, \\ A_\omega f + N_f \leqslant 2^{n-1},\ \omega \neq 0, \\ -A_\omega f + N_f \leqslant 2^{n-1},\ \omega \neq 0, \\ f \in \{0,1\}^n, N_f \geqslant 0, N_f - \text{целое}. \end{gathered} \tag{3}$$

Кроме того, с помощью целочисленного программирования можно решать задачи максимизации нелинейности несбалансированных функций и корреляционно иммунных функций. Для этого достаточно убрать из задачи 3 первые два ограничения или только первое ограничение (в зависимости от того, какую именно задачу мы решаем) и

добавить два других ограничения, полученные из формулы обращения преобразований. Новые ограничения выглядят слудующим образом:

$$A_{\mathbb{O}}f + N_f \leqslant 2^n,$$
$$-A_{\mathbb{O}}f + N_f \leqslant 0.$$

Проиллюстрируем задачу на простом примере. Рассмотрим вопрос максимизации нелинейности 1-устойчивой функции ($m = 1$) от пяти переменных ($n = 5$). Получим следующую задачу:

$$N_f \to sup$$
$$A_{(00000)}f = 2^{n-1} = 16,$$
$$A_{(00001)}f = 0,$$
$$\dots$$
$$\dots$$
$$\dots$$
$$A_{(10000)}f = 0,$$
$$A_{\omega}f + N_f \leqslant 16,\ wt(\omega) \geqslant 1,$$
$$-A_{\omega}f + N_f \leqslant 16,\ wt(\omega) \geqslant 1.$$

Решением этой задачи является функция 4966F18C с $N_f = 12$.

Пазалику и Йоханссону с помощью этого метода удалось посчитать нелинейности при пяти и шести переменных и при устойчивости порядка от единицы до пяти. При семи переменных нелинейности им посчитать не удалось ([1]). Полученные результаты можно посмотреть в таблице, приведенной ниже.

	Устойчивость					
n	0	1	2	3	4	5
5	12	12	8	0	0	0
6	26	24	24	16	0	0
7	*	*	*	*	*	0

Таблица 2. Максимальная N_f, полученная из решения 0–1 задачи ЦЛП. Символ * означает, что нелинейность посчитать не удалось.

Известен факт, что при $n = 7$ и устойчивости порядка $m \in -1 : 1$ нелинейность равна 56 ([4]). Дополним таблицу 2, посчитав максимальные нелинейности при $n = 7$ и при $m \in 2 : 5$ с помощью средств MATLAB. Напомним, что через (-1)-устойчивую функцию мы обозначаем любую булеву функцию.

	Устойчивость						
n	-1	0	1	2	3	4	5
7	56	56	56	56	48	32	0

Таблица 3. Максимальная N_f при 7 переменных, полученная из решения 0–1 задачи ЦЛП.

Приведем функции, достигающие соответствующие нелинейности при определенном порядке устойчивости.

- D9045715901872031F6894835A11B2FF — имеет -1-устойчивость и нелинейность 56;
- 5261B1504B7D196488F1DF93E31739F8 — имеет 0-устойчивость и нелинейность 56;
- A77567948BE1C94100E2FA9C6D37839E — имеет 1-устойчивость и нелинейность 56;
- 92564DE3E13DAB48BE294678856C719B — имеет 2-устойчивость и нелинейность 56;
- BC16C29743BC3DC216E99768E943683D — имеет 3-устойчивость и нелинейность 48;
- A5965A69695A96A55A69A59696A5695A — имеет 4-устойчивость и нелинейность 32;

По прошествии двух часов непрерывной работы на компьютере с процессором Intel(R) Pentium(R) CPU B980, 2.4 GHz и оперативной памятью 4 ГБ при восьми переменных и выше программа на MATLAB'е не смогла найти планы этой задачи.

5 Приближенные алгоритмы поиска булевых функций

5.1 Генетический алгоритм (GA)

Генетический алгоритм представляет из себя метод решения оптимизационных задач с ограничениями и без ограничений, основанный на процессе естественного отбора, тем самым имитируя биологическую эволюцию. Алгоритм многократно модифицирует популяцию индивидуальных решений. На каждом шаге генетического алгоритма случайным образом выбираются особи из текущей популяции и используются в качестве родителей, чтобы произвести потомков для следующего поколения. В течение последующих поколений популяция «эволюционирует» в сторону оптимального решения.

В следующей таблице кратко изложим отличия генетического алгоритма от классических оптимизационных алгоритмов:

Классический алгоритм	**Генетический алгоритм**
Генерируется одна точка на каждой итерации. Последовательность таких точек сходится к оптимальному решению.	Генерируется популяция точек на каждой итерации. Лучшая точка в популяции приближается к оптимальному решению.
Выбирается следующая точка в последовательности путем определенных в алгоритме вычислений.	Выбирается следующая популяция путем вычислений, которые используют генерацию случайных чисел.

Генетический алгоритм можно использовать для нахождения булевых функций максимальной нелинейности. Для этой задачи возьмем в качестве целевой функции функцию абсолютного максимума среди всех коэффициентов Уолша–Адамара булевой функции от n переменных. Нам необходимо минимизировать эту функцию.

Запишем целевую функцию в аналитическом виде через коэффициенты Фурье. Для этого воспользуемся формулой обращения преобразований. Мы знаем по определению, что $F_f(\omega) = A_\omega f$, где A_ω — это строка матрицы Уолша–Адамара A, соответствующая ω. Таким образом, целевая функция принимает вид:

$$g(x) = \max\{|2^{n-1} - A_{\mathbb{O}} f|, |A_{\omega_1} f|, |A_{\omega_2} f|, \ldots, |A_{\omega_{2^n-1}} f|\},$$

где $\omega_1, \ldots, \omega_{2^n-1}$ — различные ненулевые вектора длины n.

Заметим, что такая задача без ограничений на сбалансированность и порядок устойчвости. Следовательно, результатом генетического алгоритма может быть несбалансированная и не корреляционно иммунная функция. Если дополнительно наложим на эту задачу первые два ограничения из задачи 3, то мы сможем найти сбалансированные и устойчивые функции высокой нелинейности.

Благодаря генетическому алгоритму были получены следующие результаты при ограничении на количество популяций, равным $100 \cdot n$ (были использованы средства MATLAB):

	Устойчивость						
n	-1	0	1	2	3	4	5
8	112	110	108	104	98	96	94
9	234	230	226	220	218	214	210
10	474	472	460	452	448	448	444
11	966	964	936	934	922	*	*
12	1962	1950	1924	*	*	*	*

Таблица 4. Нелинейности, полученные генетическим алгоритмом[2].

5.2 Алгоритм направленного поиска (DSA)

Далее приведём универсальный алгоритм поиска лучших булевых функций по различным критериям, как, например, нелинейность, корреляционная иммунность и т.д. Авторами он был назван как **алгоритм направленного поиска (DSA)**. Алгоритм примечателен тем, что на каждой итерации он уменьшает максимальный по модулю коэффициент Уолша–Адамара, тем самым максимизируя нелинейность функции от n переменных.

Кратко опишем этот алгоритм. Пусть дана функция $f(x)$. Обозначим $\hat{f}(x) = (-1)^{f(x)}$. Пусть $\hat{f}^*$ получается из $\hat{f}$ инвертированием (в данном случае, заменой 1 на -1 или наоборот) одной из компонент, скажем, k-ой. В векторной форме такая операция может быть рассмотрена как простое суммирование векторов:

$$\hat{f}^* = \hat{f} + f^k,$$

где f^k — это вектор, который полностью состоит из нулей, за исключением k-ой позиции, где стоит 2 или -2. Инвертирование k-ой компоненты в $\hat{f}$ изменит каждый элемент вектора W_f на константу $+2$ или -2. Обозначим через W_{f^*} вектор коэффициентов Уолша–Адамара после инвертирования в $\hat{f}$. Он получен таким образом:

$$W_{f^*} = A\hat{f}^* = A\hat{f} + Af^k = W_f + A_{\cdot k} f_k,$$

где $A_{\cdot k}$ — это k-ый столбец матрицы Адамара A.

Пусть на m-ой координате вектора W_f стоит максимальный по модулю элемент (назовем его W_{max}). То есть $W_{max} = |W_f[m]| = \max_{1 \leqslant l \leqslant 2^n} |W_f[l]|$.

Введем вектор c длины 2^n, состоящий из 0 и 1. Элемент c_j равен 1, если W_{max} уменьшается на 2 после инвертирования компоненты $\hat{f}_j$. В противном случае элемент c_j равен 0. Следовательно, возьмем такие индексы $s_j, j \in 1 : p$, при которых $c_{s_j} = 1$. Любой из элементов $\hat{f}_{s_1}, \ldots, \hat{f}_{s_p}$ можно инвертировать.

W_{max} гарантированно уменьшится после инвертирования в $\hat{f}$, если W_{max} достигается только на m-ой координате. Но если более чем один элемент в W_f равен W_{max}, то не факт, что после инвертирования все такие коэффициенты уменьшатся. Кроме того, компоненты в W_f со значением $W_{max} - 2$ будут действовать непредсказуемо, то есть будут увеличиваться или уменьшаться в случайном порядке.

[2]Символ * означает, что по прошествии двух часов непрерывной работы на компьютере с процессором Intel(R) Pentium(R) CPU B980, 2.4 GHz и оперативной памятью 4 ГБ программа на MATLAB'е не смогла найти функции, удовлетворяющие ограничениям.

Чтобы избежать таких колебаний, алгоритм сортирует элементы $|W_f[i]|$ по убыванию, а затем выбирает такой индекс k, где после инвертирования элемента $\hat{f}_k$ уменьшается как можно большее количество $|W_{f_m}|$. Затем алгоритм продолжается с той же процедурой, инвертируя биты в таблице истинности новой функции. Если возникает случай, когда появляется $f(x)$, которая была на предыдущих шагах, то мы возвращаемся в начало алгоритма. Такой случай называется циклом.

Алгоритм направленного поиска может быть описан следующим образом:

1. Генерируем случайную сбалансированную булеву функцию $f(x)$.
2. Вычисляем W_f и сортируем индексы в W_f так: $|W_{f_{i_1}}| \geqslant |W_{f_{i_2}}| \geqslant \cdots \geqslant |W_{f_{i_{2^n}}}|$
3. Находим индекс s, при котором максимизируется количество $|W_{f_{i_j}}|$, которые будут уменьшаться, когда s-й бит инвертирован. Инвертируем s-й бит в $f(x)$.
4. Проверим нелинейность нового $f(x)$. Если цикл обнаружен (то есть получена функция, которая была на предыдущих шагах), переходим к шагу 1, в противном случае — к шагу 2.
5. Выведем лучшую полученную функцию.

Результирующая функция может быть не сбалансированной и не корреляционно иммунной порядка 1, но ее можно превратить.

Пусть $f(x)$ — несбалансированная функция на $\mathcal{F}_n$. Предположим, что $W_f(\omega_0) = 0$ для некоторого вектора $\omega_0 \neq 0$ в V_n. Тогда определим новую функцию $f^*(x) = f(x) \oplus \oplus \langle x, \omega_0 \rangle$ на $\mathcal{F}_n$, которая сбалансирована, имея ту же самую нелинейность и алгебраическую степень, как и $f(x)$. Очевидно, что нелинейность и алгебраическая степень остаются такими же, так как функция $f^*(x)$ отличается от $f(x)$ только в линейных строках. Ссылаясь на определение преобразования Уолша–Адамара, мы имеем,

$$W_{f^*}(0) = \sum_x (-1)^{f^*(x)} = \sum_x (-1)^{f(x)} (-1)^{\langle \omega_0, x \rangle} = W_f(\omega_0) = 0.$$

Теперь рассмотрим, как получить корреляционно иммунную функцию порядка 1. При фиксированной $f(x)$ обозначим $W^0 = \{\omega : W_f(\omega) = 0, \omega \in V_n\}$. Пусть B — матрица $m \times n, m \leqslant n$, чьи строки это линейно независимые вектора из W^0. Очевидно, если $m = n$, то достаточно переместить вектора из B в желаемые позиции, используя линейную матрицу преобразований C. Булева функция $f(x)$, для того, чтобы быть корреляционно иммунной порядка 1, должна иметь нули в $W_f(\omega)$ для $wt(\omega) = 1$. Следовательно, мы по сути перемещаем некоторые точки из W^0 в такие n точек, где $wt(\omega) = 1$, используя линейное преобразование.

Пусть $m = n$ и $C = B^{-1}$. Тогда определим новую функцию $f^*(x) = f(C \cdot x)$.

Докажем, что $f^*(x)$ является корреляционно иммунной функцией порядка 1:

$$W_{f^*}(\omega) = \sum_x (-1)^{f(C \cdot x)} (-1)^{\langle \omega, x \rangle} = \sum_x (-1)^{f(x)} (-1)^{\langle \omega, B \cdot x \rangle} = 0,$$

$$wt(\omega) = 1.$$

Таким образом, функцию, полученную в результате работы алгоритма, можно превратить в сбалансированную и корреляционно иммунную порядка 1 функцию, сохраняя

нелинейность и другие свойства. Превратить функцию в 1-устойчивую функцию не получится. Этот факт подтверждает следующий контрпример.

Пусть результатом алгоритма является функция f от пяти переменных со значениями 1141093F нелинейности 12 (максимальная). Её коэффициенты Уолша-Адамара равны $(8, 8, 8, -8, 8, 0, 0, 8, 8, 0, 0, -8, 0, 0, 0, 0, 8, 8, 0, 0, -8, 0, -8, 0, -8, 0, 8, 0, 0, 0, 8, -8)$. Превратим эту функцию в сбалансированную. Возьмем, например, $\omega_0 = (0, 1, 1, 0, 1)$. Видно, что $W_f(\omega_0) = 0$. Тогда определим функцию $f^*(x) = f(x) \oplus \langle x, \omega_0 \rangle$. Значения этой новой функции будут такими: $f^* =$ 4BE4539A. Очевидно, что новая функция сбалансирована. Более того, она является 1-устойивой.

Попробуем превратить исходную функцию f в корреляционно имунную функцию порядка 1. Возьмем пятерку векторов $\omega_1 = (0, 0, 1, 0, 1), \omega_2 = (0, 0, 1, 1, 0), \omega_3 = (0, 1, 0, 0, 1), \omega_4 = (1, 0, 0, 1, 1), \omega_5 = (1, 1, 0, 1, 1)$, которые, очевидно, принадлежат W_0. Построим из этой тройки матрицу B:

$$B = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

Соответственно, матрица $C = B^{-1}$ выглядит так (предварительно заменим -1 на 1 и 2 и -2 на 0):

$$C = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

Определим новую функцию $f^*(x) = f(C \cdot x)$. Значения этой новой функции будут $f^* =$ 6944243A, а ее коэффициенты Уолша–Адамара равны $W_{f^*} = (8, 0, 0, 8, 0, 0, 8, 8, 0, 0, 0, 0, 0, -8, 0, 8, 0, 8, -8, 0, 0, 0, -8, 8, -8, -8, 8, -8, 0, 8, 0, 8)$. Видно, что новая функция корреляционно иммунна порядка 1. Теперь осталось превратить f^* в 1-устойчивую функцию, то есть сделать из f^* сбалансированную функцию, сохранив корреляционную иммунность.

Возьмем вектор $\omega_0 = (1, 0, 0, 1, 1)$. Видно, что $W_{f^*}(\omega_0) = 0$. Тогда определим функцию $f^{**}(x) = f^*(x) \oplus \langle x, \omega_0 \rangle$. Значения этой новой функции будут $f^{**} =$ 0F22BDA3, а ее коэффициенты Уолша–Адамара равны $W_{f^{**}} = (0, -8, 8, 0, 8, -8, 0, 0, -8, 8, -8, -8, 8, 0, 8, 0, 8, 0, 0, 8, 8, 8, 0, 0, 0, 0, 0, 0, 8, 0, -8, 0)$. Новая функция сбалансирована, но не является 1-устойчивой. К сожалению, описанным выше способом нам не удалось сохранить корреляционную иммунность. Таким образом, получить 1-устойчивую функцию такими методами не всегда возможно.

5.2.1 Примеры работы алгоритма

Рассмотрим работу алгоритма на простом примере. Мы хотим вывести функцию максимальной нелинейности.

Пример. Пусть f — начальная сбалансированная функция от двух переменных со значениями $(1, 0, 0, 1)$. Посчитаем её коэффициенты Уолша–Адамара. Они равны $(0, 0, 0, -4)$. Сортируем коэффициенты по модулю: $4, 0, 0, 0$. Затем инвертируем каждый бит. Проверим все возможные варианты.

1. $f^* = (0,0,0,1).W_{f^*} = (2,2,2,-2)$. Сортируем коэффициенты по модулю: $2,2,2,2$. Один коэффициент Уолша–Адамара уменьшился.

2. $f^* = (1,1,0,1).W_{f^*} = (-2,2,-2,-2)$. Сортируем коэффициенты по модулю: $2,2,2,2$. Один коэффициент Уолша–Адамара уменьшился.

3. $f^* = (1,0,1,1).W_{f^*} = (-2,-2,2,-2)$. Сортируем коэффициенты по модулю: $2,2,2,2$. Один коэффициент Уолша–Адамара уменьшился.

4. $f^* = (1,0,0,0).W_{f^*} = (2,-2,-2,-2)$. Сортируем коэффициенты по модулю: $2,2,2,2$. Один коэффициент Уолша–Адамара уменьшился.

У нас есть четыре варианта инвертирования. Например, инвертируем в f второй бит. Посчитаем нелинейность полученной функции. Она равна 1, и она максимальна. Значит, алгоритм можно завершить.

5.2.2 Программа, реализующая алгоритм

Теперь опишем, как работает программа, реализующая алгоритм направленного поиска. Программа была написана на языке PASCAL. Рассмотрим ее работу на нахождение функции с максимальной нелинейностью.

Сначала создается текстовый файл, в котором будут находиться проверенные функции. Если там была какая-либо информация, то она уничтожается. Далее вводим с клавиатуры количество переменных n. Потом считаем количество значений функции $p = 2^n$ и максимальную нелинейность $linmax$. Максимальную нелинейность мы находим по гипотезе 1. Формулировка гипотезы следующая:

Гипотеза 1. *Максимальную нелинейность 1-устойчивой булевой функции $f(x)$ от n переменных можно найти из системы:*

$$N_f = \begin{cases} 2^{n-1} - 2^{\frac{n}{2}}, & n \text{ четное;} \\ 2^{n-1} - 2^{\frac{n-1}{2}}, & n \text{ нечетное.} \end{cases}$$

Гипотеза подтверждается результатами решения задачи целочисленного программирования при $n \in 4:7$. Такими же доводами руководствовались Пазалик и Йоханссон (см. [1]).

Теперь рассмотрим работу цикла, который реализует все шаги алгоритма направленного поиска. Сначала с помощью процедуры *balance* генерируем случайную сбалансированную булеву функцию. Создается строка, состоящая из идущих подряд единиц и нулей, количество которых одинаково. Длина этой строки равна p. Далее случайным образом выбираем индекс строки и записываем соответствующее ему число в функцию f по порядку. Удаляем взятое число из строки. Затем снова выбираем случайный индекс строки. И так далее. Записываем полученную функцию в файл. Считаем коэффициенты Уолша-Адамара полученной функции, помещаем их в модули и сортируем.

Инвертируем каждый бит функции, посчитаем коэффициенты Уолша-Адамара каждой полученной функции, аналогичным образом их упорядочим и запишем результаты. Теперь будем сравнивать коэффициенты Уолша-Адамара старой функции и инвертированных функций. Выбираем такую инвертированную функцию, у которой количество уменьшенных коэффициентов максимально.

А теперь надо проверить, была ли эта функция на предыдущих шагах или нет. Для этого прогоняем её по функциям, которые записаны в файле. Если такая функция уже была, то записываем в логическую переменную $flag$ значение $false$, в противном случае записываем значение $true$.

Если значение $flag$ является ложью, то возвращаемся к шагу 1, то есть заново генерируем случайную булеву функцию. Если значение $flag$ является истиной, то считаем её нелинейность и сравниваем с максимальной нелинейностью $linmax$. Если она равна $linmax$, то выводим результирующую функцию и её нелинейность и завершаем работу алгоритма. В противном случае возвращаемся в начало цикла.

5.3 Результаты алгоритма

При пяти переменных были получены такие функции:

- BF7931AD — функция не сбалансирована, не корреляционно иммунна и имеет нелинейность 12.
- E3625D4A — функция сбалансирована, не корреляционно иммунна и имеет нелинейность 12.

При шести переменных были получены такие функции:

- 28D319895790FECE — функция сбалансирована, не корреляционно иммунна и имеет нелинейность 24.
- 9DB12186B2D61505 — функция не сбалансирована, не корреляционно иммунна и имеет нелинейность 24.

Заключение

В работе были представлены и проанализированы некоторые результаты по нелинейности устойчивых булевых функций в определенных случаях. Были разобраны эффективные методы максимизации нелинейности при наличии линейных ограничений и построения устойчивых функций с высокой нелинейностью. Во второй части статьи были рассмотрены генетический алгоритм и алгоритм направленного поиска. Алгоритм направленного поиска предложен как очень эффективный в поиске высоко нелинейных булевых функций, которые являются сбалансированными или корреляционно иммунными порядка 1. Часть алгоритмов были запрограммированы. Также были приведены примеры функций и результаты описанных методов.

Список литературы

[1] E. Pasalic, T. Johansson «Further Results on the Relation Between Nonlinearity and Resiliency for Boolean Functions» 7th IMA International Conference In Cryptography and Coding / Lecture Notes in Computer Science, 1746, 10 p., 1999.

[2] S. Chee, S. Lee, D. Lee, S.H. Sung «On the Correlation Immune Functions and Their Nonlinearity» Advances in Cryptology — ASIACRYPT '96, Lecture Notes in Computer Science, 1163, 235–242 pp., Springer–Verlag, 1996.

[3] В.Н. Малозёмов «Дискретные функции Уолша» Семинар по дискретному гармоническому анализу и геометрическому моделированию «DHA & CAGD», 10 с., 2011.

[4] О.А. Логачев, А.А. Сальников., В.В. Ященко. «Булевы функции в теории кодирования и криптологии» М.: Московский центр непрерывного математического образования, 303–314 сс., 2004.

[5] S. Maitra, E. Pasalic «Further Constructions of Resilient Boolean Functions With Very High Nonlinearity» IEEE Transactions on Information Theory, Vol. 48, No. 7, 1825–1834 pp., 2002.

[6] Ю.В. Таранников. «Комбинаторные свойства дискретных структур и приложения к криптологии» Москва, Издательство МЦНМО, 141–142 сс., 2014.

[7] MATLAB Help «Genetic Algorithm». The Mathworks, Inc. Версия MATLAB — R2015b.