пертная оценка преимуществ и построение матриц преимуществ с помощью метода парных сравнений, рассчитан вектор глобальных приоритетов и принято решение на основе результатов анализа.
Ключевые слова: системный анализ, многокритериальный выбор, альтернативные варианты оптимизации, фаззер, система защиты.

SYSTEM ANALYSIS OF MULTICRITERION OPTIMIZATION PROBLEM FOR INFORMATION SYSTEM PROTECTION (HACKING)

The article discusses the solution of the system analysis multicriterion optimization problem and application of organizational and analytical methods development, justification and decision-making for solving the problem of choosing the means of fuzzing using genetic algorithms. Mathematical methods of analysis of expert assessments, method of analysis of hierarchies and the method of paired comparisons are applied. A systematic analysis of the fuzzing system using genetic algorithms, the search for the sources of the corresponding type system, compiling a comparative table of the prototype with known systems of this type, a systematic analysis of the problems of multicriteria selection, expert evaluation of the advantages and construction of the matrices of the advantages of using the method of pairwise comparisons, the calculated vector of global priorities and decision on the basis of the results of the analysis are represented in the article.
Index terms: system analysis, multicriterion optimization problem, alternative optimization options, fazzer, protection system.

Коваленко Юлия Борисовна, кандидат педагогических наук, доцент кафедры безопасности информационных технологий Национального авиационного университета.
E-mail: yleejulee22@gmail.com.
Коваленко Юлія Борисівна, кандидат педагогічних наук, доцент кафедри безпеки інформаційних технологій Національного авіаційного університету.
Yulia Kovalenko, Ph.D.Academic Department of IT-Security of National Aviation University.

УДК 621.391:519.7

ШВИДКІ АЛГОРИТМИ ПОБУДОВИ k -ВИМІРНИХ НАБЛИЖЕНЬ БУЛЕВИХ ФУНКЦІЙ

Антон Олексійчук, Сергій Конюшок, Артем Сторожук

Знаходження наближень булевих функцій у певних класах функцій, що мають більш просту будову, є традиційною задачею симетричної криптографії. Зокрема, при побудові кореляційних атак на потокові шифри потрібно знаходити наближення булевих функцій від n змінних k -вимірними функціями, тобто такими, що є афінно еквівалентними функціям від $k<n$ змінних. Основним результатом статті є алгоритм побудови списку всіх k -вимірних функцій степеня не вище d , які знаходяться на відносній відстані не більше $2^{-d}(1-\varepsilon)$ від булевої функції n змінних, що задасться вектором її значень, $1\le d\le k<n$, $\varepsilon\in(0,1)$. Запропонований алгоритм є більш ефективним у порівнянні з найкращим раніше відомим (у певних випадках – в 1000 та більше разів) і може бути застосований на практиці при дослідженні кореляційних властивостей функцій ускладнення потокових шифрів.
Ключові слова: *потоковий шифр, нелінійний криптоаналіз, кореляційна атака, k -вимірна булева функція, швидкий алгоритм, знаходження наближень булевих функцій.*

Вступ. Як відомо, стійкість сучасних потокових шифрів відносно кореляційних атак визначається наявністю або відсутністю наближень функцій ускладнення, що використовуються в їх конструкціях, більш просто збудованими функціями. Найвідомішими прикладами булевих функцій, які мають просту аналітичну будову, є афінні функції, а також функції, що залежать від малої кількості змінних. Більш широкий клас утворюють k -вимірні функції, тобто булеві функції від довільної кількості n змінних, що є лінійно еквівалентними функціям від фіксованого числа $k<n$ змінних. Дослідженню властивостей таких функцій, зокрема, як можливих наближень довільних булевих функцій, присвячено роботи [1, 3, 4, 8, 10, 14 – 16]. Відомі також різноманітні атаки на генератори гами потокових шифрів, функції ускладнення яких є k -вимірними або близькими до таких [2, 5, 12, 13].

Для побудови ефективних атак на потокові шифри необхідно знаходити k -вимірні функції, що є близькими до заданої функції f від n

змінних, причому, як правило, k є не надто великим числом. Ефективність розв'язання цієї задачі суттєво залежить від відносної відстані між функцією f та її шуканими наближеннями. Якщо ця відстань не перевищує $2^{-(k+1)}(1-\varepsilon)$, де $\varepsilon \in (0,1)$, існує не більше однієї шуканої функції, для знаходження якої відомі ефективні алгоритми [8, 16]. Суттєво більш складною задачею є побудова списку всіх k-вимірних функцій степеня не вище d, які знаходяться на відносній відстані не більше $2^{-d}(1-\varepsilon)$ від заданої булевої функції n змінних, $1 \le d \le k < n$, $\varepsilon \in (0,1)$. В роботі П. Гопалана [15] показано, що кількість зазначених функцій обмежена зверху величиною, яка не залежить від n, та запропоновано найефективніший на сьогодні алгоритм їх побудови.

В даній статті пропонується алгоритм, який дозволяє будувати k-вимірні наближення булевих функцій більш ефективно в порівнянні з алгоритмом Гопалана (у певних випадках – в 1000 та більше разів). Запропонований алгоритм базується на окремих результатах статті [4], які дозволяють встановити більш точну (в порівнянні з [15]) оцінку кількості шуканих наближень, а також на детальному аналізі структури таких наближень, що надає можливість помітно скоротити часову складність їх побудови.

Решта статті має такий вигляд. В п. 1 викладено теоретичні результати та допоміжні (базові) алгоритми, що використовуються при розв'язанні основної задачі. Вирішенню останньої присвячено п. 2, де представлено також результати порівняння запропонованого алгоритму з алгоритмом Гопалана. В п. 3 наведено приклад практичного застосування запропонованого алгоритму, а в завершальній частині статті сформульовано стислі висновки.

1. Постановка задачі, теоретичні результати та базові алгоритми. Нижче використовуються такі позначення:

V_n – векторний простір двійкових векторів довжини n;

$F_{n\times k}$ – множина матриць розміру $n\times k$ над полем $F = \mathbf{GF}(2)$;

$C(A)$ – підпростір векторного простору V_n, породжений стовпцями матриці $A \in F_{n\times k}$;

B_n – множина булевих функцій від n змінних;

$\deg g$ – степінь поліному Жегалкіна функції $g \in B_n$;

$d(f,g) = 2^{-n}|\{x \in V_n : f(x) \ne g(x)\}|$ – відносна відстань між функціями $f, g \in B_n$;

$\hat{f}(\alpha) = 2^{-n}\sum_{x\in V_n}(-1)^{f(x)\oplus \alpha x}$, $\alpha \in V_n$ – нормовані коефіцієнти Уолша-Адамара функції $f \in B_n$.

Для будь-якої функції $g \in B_n$ покладемо $I_g = \{\alpha \in V_n \mid \forall x \in V_n : g(x\oplus\alpha) = g(x)\}$ та позначимо $I_g^{\perp}$ підпростір, дуальний до векторного простору I_g над полем F (див., наприклад, [6]).

Функція g називається k-*вимірною*, якщо вона може бути представлена у вигляді

$$g(x) = \phi(xA),\ x \in V_n, \qquad (1)$$

де $\phi \in B_k$, $A \in F_{n\times k}$, та *строго* k-*вимірною*, якщо k є найменшим невід'ємним цілим числом, для якого існує представлення функції g у вигляді (1). Кожне таке представлення, що відповідає найменшому можливому значенню $k \in \overline{0,n}$, називається *незвідним представленням* функції g [4, 15, 16].

Позначимо $B_{n,k}$ множину k-вимірних функцій від n змінних, $\bar{B}_{n,k} = B_{n,k} \setminus B_{n,k-1}$. Для будь-яких $f \in B_n$, $\varepsilon \in (0,1)$, $d, k \in \mathbf{N}$, де $d \le k < n$, покладемо

$$B_{n,k,d}(f;\varepsilon) = \{g \in B_{n,k} : d(f,g) \le 2^{-d}(1-\varepsilon), \deg g \le d\}, \qquad (2)$$

$$\bar{B}_{n,k,d}(f;\varepsilon) = \{g \in \bar{B}_{n,k} : d(f,g) \le 2^{-d}(1-\varepsilon), \deg g \le d\}. \qquad (3)$$

Потрібно розробити алгоритм, який будує множину (2) за вектором значень функції f та числами d, k і ε.

Повне вирішення цієї задачі викладено в наступному пункті. Даний пункт присвячено розв'язанню окремої підзадачі, яка полягає в розробці алгоритмів побудування множини (3) для випадків, коли $d < k$ та $d = k$ відповідно. Алгоритми, що пропонуються, базуються на низці тверджень, які наведені нижче.

Перше з них є основним та випливає з теореми 4 і леми 5 у [4].

Твердження 1. Нехай:

$$\mu_0 = \max\{2^{1-k}\varepsilon,\ \frac{4}{3\sqrt{3}}2^{-k/2-d/2}\varepsilon^{3/2}\},$$

$$S_f(\mu_0) = \{\alpha \in V_n : |\hat{f}(\alpha)| \ge \mu_0\}. \qquad (4)$$

Тоді кожна функція $g \in \bar{B}_{n,k,d}(f;\varepsilon)$ задовольняє умові

$$I_g^{\perp} = \left\langle \{x \in I_g^{\perp} : x \in S_f(\mu_0)\} \right\rangle; \qquad (5)$$

іншими словами, векторний простір $I_g^{\perp}$ породжується всіма векторами $x \in I_g^{\perp}$, які належать множині (4).

Наступне твердження є безпосереднім наслідком твердження 2 в [1] (див. також [4], наслідок 1).

Твердження 2. Представлення (1) є незвідним тоді й тільки тоді, коли $\operatorname{rank} A = k$ та $I_\phi = \{0\}$. При цьому $I_g = \{x \in V_n : xA = 0\} = C(A)^\perp$.

Твердження 3. Нехай $g \in \overline{B}_{n,k,d}(f;\varepsilon)$. Тоді функція g має незвідне представлення вигляду (1), в якому стовпці $\alpha_1, ..., \alpha_k$ матриці A належать множині (4).

Доведення. Зафіксуємо будь-яке незвідне представлення функції g: $g(x) = \phi'(xA')$, $x \in V_n$, де $\phi' \in B_k$, $A' \in F_{n\times k}$,

$$I_{\phi'} = \{0\},\ \operatorname{rank}(A') = k. \qquad (6)$$

Згідно з твердженням 2, виконується рівність $I_g{}^\perp = C(A')$. З іншого боку, на підставі формули (5) векторний простір $I_g{}^\perp$ має базис $\alpha_1, ..., \alpha_k \in S_f(\mu_0)$.

Позначимо A матрицю, що складається з вектор-стовпців $\alpha_1, ..., \alpha_k$. Тоді $C(A) = C(A')$ і, отже, існує оборотна матриця $U \in F_{k\times k}$ така, що $A' = AU$. Покладемо $\phi(y) = \phi'(yU)$, $y \in V_k$; тоді

$$g(x) = \phi'(xA') = \phi'((xA)U) = \phi(xA),\ x \in V_n,$$

причому зазначене представлення функції g є незвідним внаслідок рівностей (6) та означення функції ϕ.

Отже, твердження доведено.

Твердження 4. Для будь-якого незвідного представлення (1) функції $g \in \overline{B}_{n,k,d}(f;\varepsilon)$ виконується співвідношення $\deg\phi = \deg g \le d$.

Доведення. Оскільки ранг матриці A у правій частині рівності (1) дорівнює k, існує оборотна матриця $W \in F_{n\times n}$ така, що $WA = \begin{pmatrix} E_k \\ 0_{n-k} \end{pmatrix}$, де E_k та 0_{n-k} – одинична та нульова матриці зазначених порядків.

Розглянемо функцію $g'(x) = g(xW)$, $x \in V_n$, лінійно еквівалентну функції g. Тоді $\deg g' = \deg g \le d$. З іншого боку, для будь-якого $x = (x_1, ..., x_n) \in V_n$ виконуються рівності

$$g'(x) = g(xW) = \phi(xWA) = \phi(x\begin{pmatrix} E_k \\ 0_{n-k} \end{pmatrix}) = \phi(x_1, ..., x_k),$$

з яких випливає, що $\deg g' = \deg\phi$. Отже, $\deg\phi = \deg g' = \deg g \le d$. Твердження доведено.

Занумеруємо зараз елементи множини $S_f(\mu_0) = \{\alpha_1, ..., \alpha_m\}$ таким чином, щоб $|\hat{f}(\alpha_1)| \ge \cdots \ge |\hat{f}(\alpha_m)|$; зауважимо, що $m \le (\mu_0)^{-2}$ (див., наприклад, [17], п. 3.2).

Позначимо $S_f(\mu_0)^{\langle k\rangle}$ сукупність усіх $n\times k$ – матриць $A = (\alpha_{i_1}, ..., \alpha_{i_k})$, які складаються з лінійно незалежних вектор-стовпців $\alpha_{i_1}, ..., \alpha_{i_k} \in S_f(\mu_0)$, що задовольняють умові $1 \le i_1 < ... < i_k \le m$. Будь-які матриці $A, A' \in S_f(\mu_0)^{\langle k\rangle}$ вважатимемо еквівалентними, якщо множини їх стовпців породжують той самий підпростір векторного простору V_n, тобто існує оборотна матриця $U \in F_{k\times k}$ така, що $A' = AU$.

Твердження 5. Нехай $A_1, ..., A_l$ – будь-яка система представників усіх класів еквівалентності матриць з множини $S_f(\mu_0)^{\langle k\rangle}$. Тоді кожна функція $g \in \overline{B}_{n,k,d}(f;\varepsilon)$ має єдине незвідне представлення вигляду $g(x) = \psi(xA_j)$, $x \in V_n$, де $j \in \overline{1,l}$. При цьому виконується нерівність $\deg\psi \le d$.

Доведення. На підставі твердження 3 існує незвідне представлення (1) функції g таке, що $A \in S_f(\mu_0)^{\langle k\rangle}$. Крім того, існує точно одна матриця A_j, $j \in \overline{1,l}$, еквівалентна матриці A.

Нехай $A = A_jU$, де U – оборотна матриця порядку k. Покладемо $\psi(y) = \phi(yU)$, $y \in V_k$. Тоді $g(x) = \phi(xA) = \phi(xA_jU) = \psi(xA_j)$, $x \in V_n$, причому останнє представлення функції g є незвідним. Звідси на підставі твердження 4 отримаємо, що $\deg\psi \le d$.

Припустимо, що поряд із наведеним, існує ще одне незвідне представлення функції g того ж самого вигляду: $g(x) = \psi'(xA_{j'})$, $x \in V_n$, де $j' \in \overline{1,l}$. Тоді стовпці кожної з матриць A_j, $A_{j'}$ породжують той самий підпростір $I_g{}^\perp$ і, отже, $j = j'$. Нарешті, оскільки $\operatorname{rank}(A_j) = k$, то з рівностей $\psi(xA_j) = \psi'(xA_j)$, $x \in V_n$, випливає, що $\psi = \psi'$.

Таким чином, функція g має єдине незвідне представлення вигляду $g(x) = \psi(xA_j)$, $x \in V_n$, де $j \in \overline{1,l}$, причому $\deg\psi \le d$, що й треба було довести.

Останнє твердження дозволяє запропонувати такий алгоритм побудови множини (3) у випадку, коли $d < k$.

Алгоритм 1. Вхід: вектор значень функції $f \in B_n$; числа $\varepsilon \in (0,1)$, $d, k \in \mathbf{N}$, де $d < k < n$.

1. Використовуючи алгоритм швидкого перетворення Адамара (див., наприклад, [6], с. 217), побудувати множину $S_f(\mu_0)=\{\alpha_1,\dots,\alpha_m\}$ вигляду (4) та впорядкувати її елементи так, щоб $|\hat{f}(\alpha_1)|\ge\dots\ge|\hat{f}(\alpha_m)|$.

2. Вибрати довільну систему $A_1,\dots,A_l$ представників усіх класів еквівалентності на множині $S_f(\mu_0)^{\langle k\rangle}$. Для будь-якого $i\in\overline{1,l}$ та кожної функції $\phi\in B_k$ такої, що $\deg\phi\le d$, $I_\phi=\{0\}$, покласти $g(x)=\phi(xA_i)$, $x\in V_n$ та перевірити умову $d(f,g)\le 2^{-d}(1-\varepsilon)$. Якщо вона виконується, включити функцію g до списку, що формується.

Результат: множина (3), яка складається з усіх функцій у сформованому списку.

Коректність алгоритму випливає безпосередньо з твердження 5.

Зауважимо, що для побудови системи $A_1,\dots,A_l$ та перевірки умови $I_\phi=\{0\}$ ($\phi\in B_k$) на другому кроці алгоритму потрібно виконати певні обчислення, обсяг яких може виявитися досить значним. Тому на цьому кроці можна організувати перебір усіх матриць $A\in S_f(\mu_0)^{\langle k\rangle}$ та функцій $\phi\in B_k$ степеня не вище d, що залежать суттєво від усіх змінних. Для кожної такої пари (A,ϕ) слід покласти $g(x)=\phi(xA)$, $x\in V_n$ та перевірити умову $d(f,g)\le 2^{-d}(1-\varepsilon)$, за виконанням якої включити функцію g до списку, що формується. Сформований таким чином список буде містити усі функції, що належать множині (3), а також деякі функції з множини (2). Зазначену **модифікацію алгоритму 1** доцільно використовувати в тому випадку, коли потужність m множини (4) є не надто великою в порівнянні з числом k. Наступне твердження дозволяє оцінити трудомісткість цього алгоритму.

Твердження 6. Трудомісткість модифікованого алгоритму 1 складає

$$T'_\varepsilon(n,k,d)=O\left(2^n nk\binom{m}{k}N_{k,d}\right) \qquad (7)$$

операцій, де $m=|S_f(\mu_0)|$,

$$\mu_0=\max\{2^{1-k}\varepsilon,\ \frac{4}{3\sqrt{3}}2^{-k/2-d/2}\varepsilon^{3/2}\},$$

$N_{k,d}=\sum_{l=0}^{k}(-1)^l\binom{k}{l}2^{\sum_{i=0}^{d}\binom{k-l}{i}}$ – кількість функцій $\phi:V_k\to\{0,1\}$ степеня не вище d, які залежать суттєво від кожної змінної.

Доведення. На першому кроці для знаходження множини (4) та впорядкування її елементів достатньо виконати $T^{(1)}=O(2^n n)$ операцій (додавання, віднімання та порівняння дійсних чисел). На другому кроці треба перебрати $\binom{m}{k}N_{k,d}$ зазначених вище пар (A,ϕ), для кожної з яких перевірити умову лінійної незалежності стовців матриці A та (за виконанням цієї умови) обчислити значення функції $g(x)=\phi(xA)$, $x\in V_n$ і перевірити нерівність $d(f,g)\le 2^{-d}(1-\varepsilon)$. Остання процедура вимагає $O(2^n nk)$ операцій (арифметичних та булевих додавань, порівнянь дійсних чисел і звернень до функції ϕ). Отже, трудомісткість другого кроку алгоритму є $T^{(2)}=O\left(2^n nk\binom{m}{k}N_{k,d}\right)$. Звідси, враховуючи рівність $T'_\varepsilon(n,k,d)=T^{(1)}+T^{(2)}$, отримаємо формулу (7). Нарешті, вираз параметра $N_{k,d}$ отримується шляхом стандартного застосування методу включення-виключення (див., наприклад, [7], с. 65). Твердження доведено.

У випадку $d=k$ можна запропонувати більш ефективний алгоритм побудови множини (3). Зауважимо, що в цьому випадку зазначена множина складається з усіх строго k-вимірних функцій, які знаходяться від функції f на відносній відстані не більше ніж $2^{-k}(1-\varepsilon)$, а параметр μ_0 дорівнює $2^{1-k}\varepsilon$.

Для кожної матриці $A\in S_f(\mu_0)^{\langle k\rangle}$ позначимо ϕ_A^* булеву функцію, що визначається за правилом

$$\phi_A^*(s)=1\ \Leftrightarrow\ \sum_{x\in V_n:xA=s}f(x)\ge 2^{n-k-1},\ s\in V_k \qquad (8)$$

та покладемо $g_A^*(x)=\phi_A^*(xA)$, $x\in V_n$. Неважко переконатися в тому (див. доведення леми 3 в [4]), що для будь-якої функції $g(x)=\phi(xA)$, $x\in V_n$ виконується нерівність $d(f,g_A^*)\le d(f,g)$. Крім того, на підставі наслідку 7 у [4] кожна така функція, що належить множині (3), відрізняється від функції g_A^* не більше ніж на одному вхідному наборі. Отже, для побудови множини (3) при $d=k$ можна використовувати такий алгоритм.

Алгоритм 2. Вхід: вектор значень функції $f\in B_n$; числа $\varepsilon\in(0,1)$, $k\in\mathbf{N}$, де $k<n$.

1. Покласти $\mu_0=2^{1-k}\varepsilon$; використовуючи алгоритм швидкого перетворення Адамара, побудувати множину $S_f(\mu_0)=\{\alpha_1,\dots,\alpha_m\}$ вигляду (4) та

впорядкувати її елементи так, щоб $|\hat{f}(\alpha_1)| \geq \cdots \geq |\hat{f}(\alpha_m)|$.

2. Для кожної матриці $A \in S_f(\mu_0)^{\langle k \rangle}$ обчислити значення функції $g_A^*(x) = \phi_A^*(xA)$, $x \in V_n$ за формулою (8) та перевірити умову $d(f, g_A^*) \leq 2^{-k}(1-\varepsilon)$. Якщо вона виконується, то

– включити функцію g до списку, що формується;

- для кожного $s \in V_k$ обчислити значення функції $g_s(x) = \phi_s(xA)$, $x \in V_n$, де $\phi_s(s) = \phi_A^*(s) \oplus 1$, $\phi_s(s') = \phi_A^*(s)$, якщо $s' \in V_k \setminus \{s\}$, та перевірити умову $d(f, g_s) \leq 2^{-k}(1-\varepsilon)$. Якщо ця умова виконується, включити функцію g_s до списку, що формується.

Результат: множина (3), яка складається з усіх функцій у сформованому списку.

Твердження 7. Трудомісткість алгоритму 2 складає

$$T_\varepsilon''(n,k) = O\left(2^{n+k} nk \binom{m}{k}\right) \qquad (9)$$

операцій, де $m = |S_f(\mu_0)|$, $\mu_0 = 2^{1-k}\varepsilon$.

Доведення. Перший крок алгоритму 2 співпадає з аналогічним кроком алгоритму 1. Отже, трудомісткість цього кроку дорівнює $T^{(1)} = O(2^n n)$ операцій (додавання, віднімання та порівняння дійсних чисел). На другому кроці треба перебрати $\binom{m}{k}$ матриць A, для кожної з яких перевірити умову лінійної незалежності її стовбців, що вимагає $O(nk^2)$ (двійкових) операцій. Далі, для кожної фіксованої матриці $A \in S_f(\mu_0)^{\langle k \rangle}$ слід перебрати не більше ніж $2^k + 1$ функцій вигляду g_A^*, g_s, де $s \in V_k$, та порівняти відносну відстань між кожною з них і функцією f з числом $2^{-k}(1-\varepsilon)$. Остання процедура вимагає $O((2^k+1)2^n nk)$ операцій (арифметичних та булевих додавань, порівнянь дійсних чисел і звернень до функції f). Отже, трудомісткість другого кроку алгоритму є $T^{(2)} = O\left(2^{n+k} nk \binom{m}{k}\right)$. Звідси, враховуючи рівність $T_\varepsilon''(n,k) = T^{(1)} + T^{(2)}$, отримаємо формулу (9). Твердження доведено.

На завершення цього пункту наведемо ще один допоміжний алгоритм, який дозволяє знайти функцію $g \in B_{n,d-1}$, розташовану від заданої функції $f \in B_n$ на відносній відстані не більше ніж $2^{-d}(1-\varepsilon)$. Відомо (див., наприклад, [16], п. 6), що існує не більше однієї такої функції, причому (за умови її існування) виконується рівність $I_g = \{\alpha \in V_n : \Delta_f(\alpha) \geq 1 - 2^{2-d}(1-\varepsilon)\}$, де

$$\Delta_f(\alpha) = 2^{-n} \sum_{x \in V_n} (-1)^{f(x \oplus \alpha) \oplus f(x)}, \ \alpha \in V_n \qquad (10)$$

є автокореляційна функція булевої функції f. Шукану функцію g можна визначити за формулою

$$g(x) = \phi_A^*(xA), \ x \in V_n, \qquad (11)$$

де A – $n \times l$ – матриця, стовпці якої утворюють базис векторного простору, дуального до підпростору I_g, $l \leq d-1$, а функція ϕ_A^* визначається за формулою (8) з заміною в ній k на l. В цілому, алгоритм знаходження функції g має такий вигляд.

Алгоритм 3. Вхід: вектор значень функції $f \in B_n$; числа $\varepsilon \in (0,1)$, $d \in \mathbf{N}$, де $2 \leq d < n$.

1. Обчислити значення (10) за допомогою алгоритму швидкого перетворення Адамара. Знайти базис $\alpha_1, \ldots, \alpha_l$ векторного простору $\{\alpha \in V_n : \Delta_f(\alpha) \geq 1 - 2^{2-d}(1-\varepsilon)\}^\perp$ за допомогою алгоритму Гаусса. Якщо $l > d-1$ або $|\{\alpha \in V_n : \Delta_f(\alpha) \geq 1 - 2^{2-d}(1-\varepsilon)\}| \neq 2^{n-l}$, закінчити роботу.

2. Сформувати з вектор-стовпців $\alpha_1, \ldots, \alpha_l$ матрицю A, визначити функцію g за формулою (11) та перевірити умову $d(f,g) \leq 2^{-d}(1-\varepsilon)$, за виконанням якої включити цю функцію до списку, що формується.

Результат: множина, що складається не більше ніж з однієї функції $g \in B_{n,d-1}$, яка задовольняє умові $d(f,g) \leq 2^{-d}(1-\varepsilon)$.

Твердження 8. Трудомісткість алгоритму 3 складає

$$T_\varepsilon'''(n,d) = O\left(2^n (n^2 + nd)\right) \qquad (12)$$

операцій.

Доведення. Для побудови множини $\{\alpha \in V_n : \Delta_f(\alpha) \geq 1 - 2^{2-d}(1-\varepsilon)\}$ на кроці 1 достатньо виконати $O(2^n n)$ операцій (додавання, віднімання та порівняння дійсних чисел), а для знаходження векторів $\alpha_1, \ldots, \alpha_l$ – ще $O(2^n n^2)$ (двійкових) операцій. Отже, трудомісткість кроку 1 складає $O(2^n n^2)$. Нарешті, оскільки трудомісткість кроку 2 дорівнює $O(2^n nl) = O(2^n nd)$, то трудомісткість всього алгоритму визначається за формулою (12). Твердження доведено.

2. Алгоритм розв'язання основної задачі та його порівняння з алгоритмом Гопалана. Викладені вище результати дозволяють запропонувати алгоритм побудови множини (2) за вектором значень функції f та числами d, k і ε. Помітимо, що ця множина є об'єднанням $k-d+2$ множин, що не перетинаються:

$$B_{n,k,d}(f;\varepsilon)=\{g\in B_{n,d-1}: d(f,g)\le 2^{-d}(1-\varepsilon)\}\cup \overline{B}_{n,d,d}(f;\varepsilon)\cup\left(\bigcup_{i=d+1}^{k}\overline{B}_{n,i,d}(f;\varepsilon)\right).$$

Отже, для побудови зазначених множин можна скористатися запропонованими вище алгоритмами.

Алгоритм 4. Вхід: вектор значень функції $f\in B_n$; числа $\varepsilon\in(0,1)$, $d,k\in\mathbf{N}$, де $2\le d\le k<n$.

1. Для кожного $i\in\overline{d+1,k}$ застосувати модифікований алгоритм 1 до вхідних даних $f,\ \varepsilon,\ d,\ i$.

2. Застосувати алгоритм 2 до вхідних даних $f,\ \varepsilon,\ d$.

3. Застосувати алгоритм 3 до вхідних даних $f,\ \varepsilon,\ d$.

4. Об'єднати множини, отримані на кроках 1, 2, 3.

Результат: множина (2), отримана на кроці 4.

Безпосередньо з тверджень 6 – 8 випливає такий результат.

Твердження 9. Трудомісткість алгоритму 4 складає

$$T^{(\varepsilon)}_{n.k.d}=O\left(2^n(n^2+nd)+2^{n+d}nd\binom{m_d}{d}+2^n n\sum_{i=d+1}^{k} i\binom{m_i}{i}N_{i,d}\right) \tag{13}$$

операцій, де

$$m_i=|S_f(\mu_{0,i})|,\ \mu_{0,i}=\max\{2^{1-i}\varepsilon,\ \frac{4}{3\sqrt{3}}2^{-i/2-d/2}\varepsilon^{3/2}\},$$
$$i\in\overline{d,k};$$
$$N_{i,d}=\sum_{l=0}^{i}(-1)^l\binom{i}{l}2^{\sum_{j=0}^{d}\binom{i-l}{j}},\ i\in\overline{d+1,k}.$$

Порівняємо трудомісткість алгоритму 4 з трудомісткістю алгоритму, запропонованого П. Гопаланом [15]. Нагадаємо, що останній полягає у формуванні множини

$S_f(\mu)=\{\alpha\in V_n:|\hat{f}(\alpha)|\ge\mu\}$, де $\mu=\frac{1}{8\sqrt{2}}2^{-k/2-d}\varepsilon^2$,

переборі усіх наборів $(\alpha_1,\dots,\alpha_k,\phi)$, де $\alpha_1,\dots,\alpha_k\in S_f(\mu)$, $\phi\in B_k$, $\deg\phi\le d$, та перевірці умови $d(f,g)\le 2^{-d}(1-\varepsilon)$ для функції $g(x)=\phi(\alpha_1x,\dots,\alpha_kx)$, $x\in V_n$. Трудомісткість цього алгоритму складає не менше ніж

$$\tilde{T}^{(\varepsilon)}_{n.k.d}=2^n m^k nk\ N(k,d) \tag{14}$$

операцій (того ж самого типу, що використовуються в алгоритмах 1 – 3), де $m=|S_f(\mu)|$, $N(k,d)=2^{\sum_{i=0}^{d}\binom{k}{i}}$ – число булевих функцій степеня не вище d від k змінних.

Як видно з формул (13), (14), трудомісткості обох алгоритмів суттєво залежать від спектру Уолша-Адамара вхідної функції f, а саме, від чисел $m_d,\dots,m_k,m$. Для порівняння ефективності алгоритмів отримаємо більш прості (але більш грубі) оцінки їх трудомісткості. Помітимо, що в силу означень параметрів $\mu_{0,i}$ ($i\in\overline{d,k}$) та μ справедливі нерівності $\mu_{0,d}>\dots>\mu_{0,k}>\mu$, з яких випливає, що $S_f(\mu_{0,d})\subseteq\dots\subseteq S_f(\mu_{0,d})\subseteq S_f(\mu_{0,d})$ і, отже, $m_d\le\dots\le m_k\le m$. Таким чином, на підставі формул (13), (14) трудомісткість алгоритму 4 не перевищує значення

$$T^{(\varepsilon)}_{n.k.d}(m_k)=O\left(2^n(n^2+nd)+2^{n+d}nd\binom{m_k}{d}+2^n n\sum_{i=d+1}^{k} i\binom{m_k}{i}N_{i,d}\right), \tag{15}$$

в той час як трудомісткість алгоритму Гопалана є не менше ніж

$$\tilde{T}^{(\varepsilon)}_{n.k.d}=2^n(m_k)^k nk\ N(k,d), \tag{16}$$

де $m_k=|\{\alpha\in V_n:|\hat{f}(\alpha)|\ge\mu_{0,k}\}|$,

$\mu_{0,k}=\max\{2^{1-k}\varepsilon,\ \frac{4}{3\sqrt{3}}2^{-k/2-d/2}\varepsilon^{3/2}\}$. Зазначимо, що обидва вирази (15), (16) залежать від єдиного параметра m_k, який визначається вхідною функцією f та може приймати цілочисельні значення від 0 до $(\mu_{0,k})^{-2}$.

В табл. 1 наведено значення параметрів (15) та (16), отримані для низки значень n, k, d, m_k і ε.

Як видно з таблиці, алгоритм 4 дозволяє будувати k -вимірні наближення булевих функцій більш ефективно в порівнянні з алгоритмом Гопалана (у певних випадках – в 1000 та більше разів) і може бути застосований на практиці при малих значеннях k і d, якщо кількість m_k «відносно великих» за модулем коефіцієнтів Уолша-Адамара функції f не перевищує 20. У випадку $d=k$ трудомісткість алгоритму 4 помітно зменшується, що робить можливим його застосування при більших значеннях k (наприклад, $k=10$), в той час як алгоритм Гопалана стає практично незастосовним.

Таблиця 1

Результати порівняння ефективностей алгоритму 4 та алгоритму Гопалана ($\varepsilon = 0{,}125$)

Параметри					$T^{(\varepsilon)}_{n.k.d}(m_k)$	$\tilde{T}^{(\varepsilon)}_{n.k.d}$
n	k	d	$\log(\mu_{0,k})^{-2}$	m_k		
10	4	2	12,00	5	$2^{28,42}$	$2^{35,61}$
				20	$2^{38,24}$	$2^{43,61}$
		3	12,00	5	$2^{32,60}$	$2^{39,61}$
				20	$2^{45,52}$	$2^{47,61}$
		4	12,00	5	$2^{21,71}$	$2^{40,61}$
				20	$2^{31,56}$	$2^{48,61}$
	5	2	18,00	6	$2^{34,10}$	$2^{44,57}$
				20	$2^{45,36}$	$2^{53,25}$
		4	18,00	6	$2^{49,23}$	$2^{59,57}$
				20	$2^{60,56}$	$2^{68,25}$
		5	18,00	6	$2^{23,25}$	$2^{60,57}$
				20	$2^{34,56}$	$2^{69,25}$
15	4	2	12,00	5	$2^{34,01}$	$2^{41,19}$
				20	$2^{43,82}$	$2^{49,19}$
		3	12,00	5	$2^{38,19}$	$2^{45,19}$
				20	$2^{48,11}$	$2^{53,19}$
		4	12,00	5	$2^{27,31}$	$2^{46,19}$
				20	$2^{37,15}$	$2^{54,19}$
	5	2	18,00	6	$2^{39,68}$	$2^{50,15}$
				20	$2^{50,94}$	$2^{58,84}$
		4	18,00	6	$2^{54,81}$	$2^{65,15}$
				20	$2^{66,15}$	$2^{73,84}$
		5	18,00	6	$2^{28,84}$	$2^{66,15}$
				20	$2^{40,15}$	$2^{74,84}$

3. Приклад застосування запропонованого алгоритму. Алгоритм 4 було реалізовано програмно та застосовано до аналізу кореляційних властивостей низки булевих функцій від невеликої кількості змінних. Як приклад, що ілюструє отримані результати, розглянемо функцію f від п'яти змінних з класу Карле-Фенга [11] (табл. 2). Вибір цієї функції обумовлено її гарними криптографічними властивостями, зокрема, високою нелінійністю (тобто відстанню до класу $B_{n,1}$).

В табл. 3 показано ненормовані коефіцієнти Уолша-Адамара даної функції, впорядковані за незростанням модулів їх значень. Зрозуміло, що при $d = k$ та $0 < \varepsilon \le 1/4$ множина $S_f(\mu_0)$ вигляду (4) складається з усіх векторів α в табл. 3, які задовольняють умові $\hat{f}(\alpha) > 0$. В цьому випадку застосування алгоритму 4 для кожного $k = 2, 3, 4$ показує відсутність k-вимірних наближень функції f, які знаходяться від неї на відносній відстані не більше ніж $2^{-k}(1-\varepsilon)$.

Поряд з тим, запропонований алгоритм дозволяє отримати інформацію про найкращі наближення (1) функції f, які задаються матрицями $A \in S_f(\mu_0)^{\langle k \rangle}$ (див. позначення перед формулюванням твердження 5) та довільними функціями $\phi \in B_k$. Результати представлено в табл. 4 (обчислення проведені на ЕОМ з процесором Intel (R) Core(TM) i5-2300 CPU @ 2.80GHz та обсягом оперативної пам`яті 4 ГБ RAM на базі 32-розрядної ОС Windows 7 Service Pack 1 в середовищі Microsoft Visual Studio 2010 (.NET Framefork 4.0, мова програмування – C#); час виконання алгоритму для кожного k складає декілька секунд).

Таблиця 2

Функція Карле-Фенга від $n=5$ змінних

x	$f(x)$	x	$f(x)$	x	$f(x)$	x	$f(x)$
00000	1	01000	1	10000	1	11000	0
00001	1	01001	1	10001	0	11001	0
00010	1	01010	0	10010	1	11010	0
00011	1	01011	1	10011	1	11011	0
00100	1	01100	1	10100	0	11100	0
00101	0	01101	0	10101	0	11101	0
00110	1	01110	0	10110	1	11110	1
00111	0	01111	1	10111	0	11111	0

Таблиця 3

Коефіцієнти Уолша-Адамара функції f

α	$32\hat{f}(\alpha)$	α	$32\hat{f}(\alpha)$	α	$32\hat{f}(\alpha)$	α	$32\hat{f}(\alpha)$
10000	– 12	01100	– 8	10101	4	10100	– 4
00101	8	10010	– 8	11000	4	11001	– 4
10111	8	10011	– 8	11100	4	00000	0
11011	8	00010	4	11101	4	01101	0
00001	– 8	01010	4	00011	– 4	10110	0
00100	– 8	01011	4	00110	– 4	11010	0
01000	– 8	01110	4	00111	– 4	11110	0
01001	– 8	10001	4	01111	– 4	11111	0

Таблиця 4

Результати дослідження k -вимірних наближень функції f

k	Відстань до найближчої функції (1) такої, що $A\in S_f(\mu_0)^{\langle k\rangle}$	Кількість наближень на цій відстані	Приклади наближень: ϕ , A^T
2	10	26	$1\oplus x_1\oplus x_2$, $\begin{pmatrix}10011\\00011\end{pmatrix}$.
3	8	68	$1\oplus x_2\oplus x_1x_2\oplus x_1x_3\oplus x_2x_3\oplus x_1x_2x_3$, $\begin{pmatrix}00011\\00111\\10100\end{pmatrix}$.
4	6	84	$1\oplus x_2\oplus x_3\oplus x_1x_3\oplus x_1x_4\oplus x_2x_4\oplus x_1x_2x_3\oplus$ $\oplus x_1x_2x_4\oplus x_1x_3x_4\oplus x_2x_3x_4\oplus x_1x_2x_3x_4$, $\begin{pmatrix}01110\\10101\\11100\\11001\end{pmatrix}$.

Висновки. Основним результатом статті є алгоритм побудови списку всіх k -вимірних функцій степеня не вище d , які знаходяться на відносній відстані не більше $2^{-d}(1-\varepsilon)$ від булевої функції n змінних, що задається вектором значень, $1\le d\le k<n$, $\varepsilon\in(0,1)$. Запропонований алгоритм є більш ефективним в порівнянні з найкращим раніше відомим алгоритмом П. Гопалана [15] (у певних випадках – в 1000 та більше разів) і може бути застосований на практиці при аналізі кореляційних властивостей функцій ускладнення потокових шифрів при малих знаяеннях k і d, якщо кількість «відносно великих» за модулем коефіцієнтів Уолша-Адамара функції f не перевищує 20 (див. табл. 1). Зменшення трудомісткості запропонованого алгоритму в порівнянні з алгоритмом Гопалана досягається за рахунок застосування більш точної оцінки кількості шуканих наближень вхідної функції (твердження 1), а також більш економної організації обчислень, яка базується на детальному аналізі структури цих наближень (твердження 3 – 5).

Як і алгоритм Гопалана, алгоритми 1 – 4 можуть бути використані для побудови k -вимірних наближень булевих функцій, що задаються за допомогою оракулів (а не тільки векторів значень). В цьому випадку на першому кроці алгоритмів 1 і 2 замість швидкого перетворення Адамара слід застосовувати один з відомих швидких алгоритмів побудови високоймовірних лінійних наближень вхідної функції, наприклад, вдосконалений алгоритм Левіна [9].

ЛІТЕРАТУРА

[1]. Алексеев Е.К. О некоторых мерах нелинейности булевых функций / Е. К. Алексеев // Прикладная дискретная математика. 2011. № 2(12). С. 5–16.

[2]. Алексеев Е.К. Об атаке на фильтрующий генератор с функцией усложнения, близкой к алгебраически вырожденной / Е.К. Алексеев // Сборник статей молодых ученых факультета МВК МГУ, 2011. Вып. 8. С. 114–123.

[3]. Алексейчук А.Н. Усовершенствованный тест k-мерности для булевых функций / А. Н. Алексейчук, С. Н. Конюшок // Кибернетика и системный анализ. 2013. Т. 49. № 2. С. 27–35.

[4]. Алексейчук А.Н. Алгебраически вырожденные приближения булевых функций / А. Н. Алексейчук, С. Н. Конюшок // Кибернетика и системный анализ. 2014. Т. 50. № 6. С. 3– 14.

[5]. Алексейчук А.Н. Статистическая атака на генератор гаммы с линейным законом реинициализации начального состояния и функцией усложнения, близкой к алгебраически вырожденной / А.Н. Алексейчук, С.Н. Конюшок, А.Ю. Сторожук // Радиотехника. 2014. Вып. 176. С. 13 – 21.

[6]. Логачев О.А. Булевы функции в теории кодирования и криптологии / О.А. Логачев, А.А. Сальников, В.В. Ященко. М. : МЦНМО, 2004. 470 с.

[7]. Сачков В.Н. Введение в комбинаторные методы дискретной математики / В.Н. Сачков. М.: МНЦНМО, 2004. 424 с.

[8]. Alekseychuk A.N. Fast algorithm for reconstruction of high-probable low-dimensional approximations for Boolean functions / A.N. Alekseychuk, S.N. Konyushok // Modern Stochastics: Theory and Applications III, Proceedings. Kyiv. Taras Shevchenko National University. 2012. P. 32.

[9]. Bshouty N. More efficient PAC-learning of DNF with membership queries under the uniform distribution / N. Bshouty, J. Jackson, C. Tamon // Proc. 12th Annual Conf. on Comput. Learning Theory, 1999. P. 286 – 295.

[10]. Canteaut A. On the correlations between a combining function and function of fewer variables / A. Canteaut // The 2002 IEEE Information Theory Workshop, Proceedings. Berlin. Springer-Verlag. 2002. P. 78–81.

[11]. Carlet C. An infinite class of balanced functions with optimal algebraic immunity, good immunity to fast algebraic attacks, and good nonlinearity / C. Carlet, K. Feng // ASIACRYPT'08, Proceedings. Berlin. Springer-Verlag, 2008. P. 425–440.

[12]. Golić J. On the resynchronization attack / J. Golić, G. Morgari // Fast Software Encryption. – FSE'03, Proceedings. Berlin. Springer-Verlag, 2003. P. 100–110.

[13]. Daemen J. Resynchronization weaknesses in synchronous stream ciphers / J. Daemen, R. Govaerts, J. Vandewalle // Advances in Cryptology. – EUROCRYPT'93, Proceedings. Berlin. Springer-Verlag, 1993. P. 159–167.

[14]. Dawson E. Construction of correlation immune Boolean functions / E. Dawson, C.K. Wu // Information and Communication Security, Proceedings. Berlin. Springer-Verlag. 1997. P. 170–180.

[15]. Gopalan P. A Fourier-analytic approach to Reed-Muller decoding / P. Gopalan // Annual IEEE Symp. on Foundation in Computer Science. – FOCS 2010, Proceedings. Berlin. Springer-Verlag. 2010. P. 685–694.

[16]. Gopalan P. Testing Fourier dimensionality and sparsity / P. Gopalan, R. O'Donnel, A. Servedio, A. Shpilka, K. Wimmer // SIAM J. on Computing. 2011. V. 40(4). P. 1075–1100.

[17]. De Wolf R. A brief introduction to Fourier analysis on the Boolean cube / R. de Wolf // Theory of Comput. Library. 2008. № 1. P. 1– 20.

REFERENCES

[1]. Alekseev E.K. (2011), «On some measures of nonlinearity for Boolean functions» (in Russian), *Prikl. Diskr. Mat.*, No. 2, pp. 5-16.

[2]. Alekseev E.K. (2011), «The attack on the filter generator with complicating function closed to algebraically degenerate function» (in Russian), *Sbornik statey molodyh uchenyh MVK MGU*, No. 8, pp. 114-123.

[3]. Alekseychuk A.N., Konyushok S.N. (2013), «An improved test of Boolean functions for k-dimensionality» (in Russian), *Kibernetika i Sistemnyi Analiz*. No. 2. pp. 27-35.

[4]. Alekseychuk A.N., Konyushok S.N. (2014), «Algebraically degenerate approximation of Boolean functions» (in Russian), *Kibernetika i Sistemnyi Analiz*. No. 6. pp. 3-14.

[5]. Alekseychuk A.N., Konyushok S.N., Storozhuk A.Y. (2014), «Statistical attack on gamma generator with linear law re-initialization of the initial state and complicating function at short distance from the algebraic degenerate function», *Radiotehnika*, No. 176, pp. 13-21.

[6]. Logachev O.A., Sal`nikov A. A, Yashchenko V.V. (2004), «Boolean functions in coding theory and cryptology» (in Russian), Moscow, MCCME, 470 p.

[7]. Sachkov V.N. «Introduction to Combinatorial methods of Discrete Mathematics» (in Russian), Moscow. MCCME, 2004. 424 p.

[8]. Alekseychuk A.N., Konyushok S.N. (2012), «Fast algorithm for reconstruction of high-probable low-dimensional approximations for Boolean functions», *Modern Stochastics: Theory and Applications III, Proceedings*, Kyiv, Taras Shevchenko National University, pp. 32.

[9]. Bshouty N., Jackson J, Tamon C. (1999), «More efficient PAC-learning of DNF with membership

queries under the uniform distribution», *Proc. 12th Annual Conf. on Comput. Learning Theory*, pp. 286-295.
[10]. Canteaut A. (2002), «On the correlations between a combining function and function of fewer variables», *The 2002 IEEE Information Theory Workshop, Proceedings*, Berlin, Springer-Verlag, pp. 78-81.
[11]. Carlet C., Feng K. (2008), «An infinite class of balanced functions with optimal algebraic immunity, good immunity to fast algebraic attacks, and good nonlinearity», *ASIACRYPT'08, Proceedings*, Berlin, Springer-Verlag, pp. 425-440.
[12]. Golić J., Morgari G. (2003), «On the resynchronization attack», *Fast Software Encryption. – FSE'03, Proceedings*, Berlin, Springer-Verlag, pp. 100-110.
[13]. Daemen J., Govaerts R., Vandewalle J. (1993), «Resynchronization weaknesses in synchronous stream ciphers», *Advances in Cryptology. – EUROCRYPT'93, Proceedings*, Berlin, Springer-Verlag, pp. 159-167.
[14]. Dawson E., Wu C. K. (1997), «Construction of correlation immune Boolean functions», *Information and Communication Security, Proceedings*, Berlin, Springer-Verlag, pp. 170-180.
[15]. Gopalan P. (2010), «A Fourier-analytic approach to Reed-Muller decoding», *Annual IEEE Symp. on Foundation in Computer Science. – FOCS 2010, Proceedings*, Berlin, Springer-Verlag, pp. 685-694.
[16]. Gopalan P., O'Donnel R., Servedio A., Shpilka A., Wimmer K. (2011), «Testing Fourier dimensionality and sparsity», *SIAM J. on Computing*, V. 40(4), pp. 1075-1100.
[17]. De Wolf R. (2008), «A brief introduction to Fourier analysis on the Boolean cube», *Theory of Comput. Library*, No. 1, pp. 1-20.

БЫСТРЫЕ АЛГОРИТМЫ ПОСТРОЕНИЯ k -МЕРНЫХ ПРИБЛИЖЕНИЙ БУЛЕВЫХ ФУНКЦИЙ

Нахождение приближений булевых функций в определенных классах функций, имеющих более простое строение, является традиционной задачей симметричной криптографии. В частности, при построении корреляционных атак на поточные шифры требуется находить приближения булевых функций от n переменных k -мерными функциями, то есть такими, которые аффинно эквивалентны функциям от $k < n$ переменных. Основным результатом статьи является алгоритм построения списка всех k -мерных функций степени не выше d, находящихся на относительном расстоянии не более $2^{-d}(1-\varepsilon)$ от булевой функции n переменных, заданной вектором ее значений, $1 \le d \le k < n$, $\varepsilon \in (0,1)$.

Предложенный алгоритм является более эффективным по сравнению с лучшим ранее известным (в некоторых случаях – в 1000 и более раз) и может быть использован на практике при исследовании корреляционных свойств функций усложнения поточных шифров.

Ключевые слова: поточный шифр, нелинейный криптоанализ, корреляционная атака, k -мерная булева функция, быстрый алгоритм, нахождение приближений булевых функций.

FAST ALGORITHMS FOR CONSTRUCTING k - DIMENSIONAL APPROXIMATIONS OF BOOLEAN FUNCTIONS

Finding Boolean functions' approximations in certain classes of functions with a simple structure, is a traditional task in symmetric cryptography. In particular, correlation attacks on stream ciphers need to find approximations of n -variable Boolean functions by k -dimensional functions, i.e., the functions, which are affine equivalent to functions of $k < n$ variables. Main result of this paper is an algorithm for constructing a list of all k -dimensional functions of degree at most d at relative distance not more than $2^{-d}(1-\varepsilon)$ from a given Boolean function of n variables, defined by the truth table, $1 \le d \le k < n$, $\varepsilon \in (0,1)$. The proposed algorithm is more efficient than the best previously known (in some cases – to 1000 times and more) and can be used in practice while studying the correlation properties of stream cipher' complicating functions.

Index terms: stream cipher, non-linear cryptanalysis, correlation attack, k -dimensional Boolean function, fast algorithm, finding approximations of Boolean functions**.**

Олексійчук Антон Миколайович, доктор технічних наук, професор Інституту спеціального зв'язку та захисту інформації НТУУ «КПІ».
E-mail: alex-dtn@ukr.net
Алексейчук Антон Николаевич, доктор технических наук, профессор Института специальной связи и защиты информации НТУУ «КПИ».
Anton Alekseychuk, Doctor of Technical Science, Professor of Institute of Special Communication and Information Security of NTUU «KPI».

Конюшок Сергій Миколайович, кандидат технічних наук, доцент, заступник начальника Інституту спеціального зв'язку та захисту інформації НТУУ «КПІ».
E-mail: 3tooth@mail.ru
Конюшок Сергей Николаевич, кандидат технических наук, доцент, заместитель начальника Института специальной связи и защиты информации НТУУ «КПИ».
Sergey Konyushok, Candidate of Technical Science, docent, vice-head of Institute of Special Communication and Information Security of NTUU «KPI».

Сторожук Артем Юрійович, аспірант Інституту спеціального зв'язку та захисту інформації НТУУ «КПІ».
E-mail: storojs72@gmail.com
Сторожук Артем Юрьевич, аспирант Института специальной связи и защиты информации НТУУ «КПИ».
Artem Storozhuk, post-graduate student of Institute of Special Communication and Information Security of NTUU «KPI».